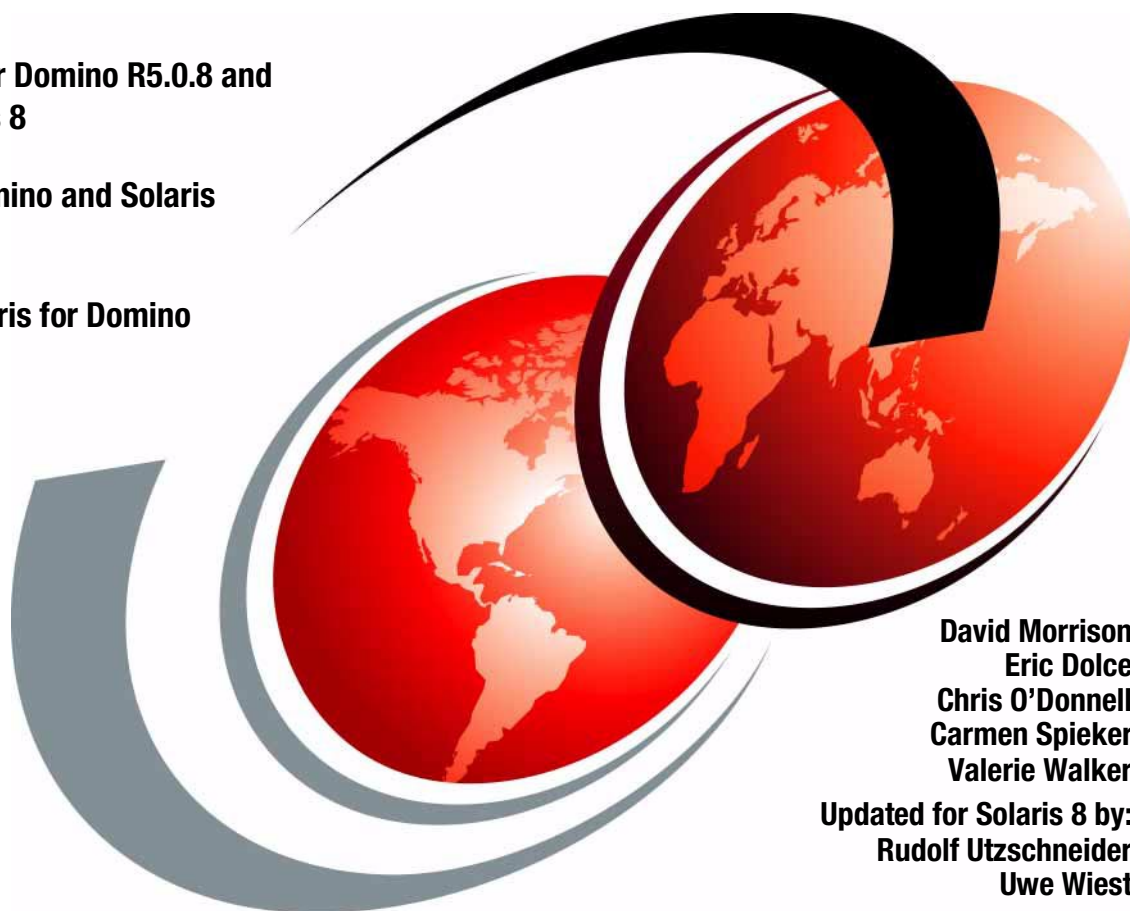


Lotus Domino R5 for Sun Solaris 8

Updated for Domino R5.0.8 and
Sun Solaris 8

Tuning Domino and Solaris

Sizing Solaris for Domino



David Morrison
Eric Dolce
Chris O'Donnell
Carmen Spieker
Valerie Walker

Updated for Solaris 8 by:
Rudolf Utzschneider
Uwe Wiest



International Technical Support Organization

Lotus Domino R5 for Sun Solaris 8

November 2001

Take Note! Before using this information and the product it supports, be sure to read the general information in “Special notices” on page 425.

Second Edition (November 2001)

This edition applies to Domino Release 5.0.8 for use on the Sun Solaris 8 Operating Environment.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. TQH 1CP-5605E
1 Charles Park
Cambridge, Massachusetts 02142-1245

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2000, 2001. All rights reserved.

Note to U.S Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Summary of changes

This section describes the technical changes made in this edition of the book and in previous editions. This edition may also include minor corrections and editorial changes that are not identified.

Second edition, November 2001

This revision reflects the addition, deletion, or modification of new and changed information described below.

New information

- ▶ Updated to include new information on Domino R5.0.8
- ▶ Updated to include new information on Sun Solaris 8
- ▶ New material: Chapter 13, “Lotus iNotes Web Access”
- ▶ New material: Appendix E, “Example TCP Port Notes.ini settings”
- ▶ New material: Appendix F, “Example script to start and shut down a Domino server”
- ▶ New material: Appendix G, “Domino IP resolve process”

Contents

Summary of changes	iii
Second edition, November 2001	iii
Preface	xv
The team that wrote this redbook	xv
Chapter 1. Introduction to Domino R5 for Sun Solaris	1
1.1 Lotus Domino R5 defined	2
1.2 Lotus Notes R5 defined	2
1.2.1 Lotus Domino R5 family of servers and clients	2
1.2.2 New terminology in Lotus Notes R5	3
1.2.3 Available platforms	4
Chapter 2. Sizing and configuration of Solaris	9
2.1 What's new in Solaris 8	10
2.1.1 Solaris 8 Admin Pack	10
2.1.2 Diagnostic and configuration tools	10
2.1.3 Installation and management	11
2.2 Sizing requirements for a Domino mail server	12
2.2.1 CPU	13
2.2.2 Memory	14
2.2.3 Disk space	14
2.2.4 Network I/O sizing	16
2.3 Solaris patch considerations	17
2.3.1 Checking for available patches	17
2.4 Operating system considerations	18
2.4.1 Disk drive configuration	19
2.4.2 Configuring /tmp (Swap)	20
2.4.3 Configuring default services	20
2.4.4 Edit Solaris system configuration (/etc/system)	21
2.4.5 Domino partitioning	22
2.4.6 Domino clustering	23
2.4.7 Hardware clustering	24
2.4.8 Network redirection	24
2.5 Configuration checklist	24
2.5.1 Creating users and groups	25
2.5.2 UNIX environment for Domino user ID	30
2.5.3 Network configuration	30
2.5.4 File system layout	36

2.6	Moving applications from NT to UNIX	37
2.6.1	Moving the application to the Solaris server	38
2.7	Summary	40
Chapter 3. Installing Domino R5 on Sun Solaris		43
3.1	Prerequisites checklist	44
3.1.1	Install and configure the Solaris environment	44
3.1.2	Prepare the Domino installation	45
3.2	Install Domino server code	46
3.3	Setting up Domino servers	57
3.3.1	Setting up the first Domino server	57
3.3.2	Setting up an additional Domino server	64
3.3.3	Rerunning the Domino server setup	73
3.4	Last configuration steps	74
3.4.1	IP address binding	74
3.4.2	Configuring partitioned servers	74
3.4.3	NRPC port to IP address binding	75
3.4.4	Setting up security for your Domino server	79
3.4.5	Removing a password from a server ID file	79
3.5	Starting the Domino server	81
3.5.1	Starting the Domino server from Solaris command line	81
3.5.2	Starting the Domino server in the background	82
3.5.3	Starting the Domino server using a startup script	83
3.6	Shutting down the Domino server	85
3.6.1	Shutting down from a foreground server console	86
3.6.2	Shutting down from the Domino Administrator	87
3.6.3	Shutting down from the Solaris command line	87
3.6.4	Shutting down the server from a script	88
3.7	Summary	89
Chapter 4. Tuning Domino Server on Solaris		91
4.1	Solaris OS considerations	92
4.1.1	Considerations for clustered servers	94
4.1.2	Solaris kernel tuning	95
4.1.3	Solaris file system tuning	95
4.2	Network configuration	95
4.2.1	TCP/IP maximum transmission unit (MTU) sizing	97
4.3	RAID	102
4.4	Domino settings	103
4.4.1	Common settings for all Domino servers	104
4.4.2	Settings for mail servers	106
4.4.3	Settings for Web clients	108
4.4.4	Settings specific to partitions and clusters	108

4.5 Summary	109
Chapter 5. Domino advanced services	111
5.1 Domino partitioning	112
5.1.1 Installation	112
5.1.2 Configuration	113
5.1.3 Configuring memory resources for partitioning	118
5.1.4 Troubleshooting	120
5.2 Domino clustering	120
5.2.1 Workload balancing	121
5.2.2 Failover	123
5.2.3 Creating the cluster	123
5.2.4 Cluster directory database	127
5.2.5 Removing a server from a cluster	127
5.2.6 Setting up your cluster	130
5.2.7 Cluster statistics	131
5.2.8 Troubleshooting	132
5.3 Billing	134
5.3.1 How it works	134
5.3.2 Configuration	134
5.3.3 Troubleshooting	136
5.3.4 Customizing billing	136
5.4 Summary	136
Chapter 6. Administration	137
6.1 The Domino Administrator client	138
6.1.1 Overview	138
6.1.2 Starting the Domino Administrator client	138
6.1.3 Using server lists	139
6.1.4 Setting administration preferences	140
6.1.5 Using tabbed pages	142
6.2 The Web Administrator	150
6.2.1 Web Administrator functionality	150
6.2.2 Web Administrator limitations	151
6.2.3 Working with the Web Administrator	151
6.3 The Domino Character Console	165
6.3.1 Starting Domino Character Console	166
6.3.2 Stopping Domino Character Console	167
6.3.3 Commands	168
6.4 Summary	168
Chapter 7. Security	169
7.1 Solaris operating environment security	170
7.2 File systems and local security	170

7.2.1	Solaris patches	170
7.2.2	File system	171
7.2.3	Mounting file systems	173
7.2.4	Accounts	174
7.2.5	Cron and at security	175
7.2.6	The init system	176
7.2.7	System default umask	177
7.2.8	Log files	177
7.2.9	The login command	178
7.3	Network service security	178
7.3.1	Telnet	179
7.3.2	Remote access services (rsh, rlogin, and rcp)	179
7.3.3	Remote execution service (rexec)	180
7.3.4	FTP	180
7.3.5	Managed services: inetd	180
7.3.6	Network File System (NFS)	181
7.3.7	Automount	181
7.3.8	Sendmail	182
7.3.9	Print services	182
7.3.10	Reducing inetsvc	182
7.3.11	The Solaris ndd command	183
7.4	Security tools	184
7.4.1	The sudo tool	184
7.4.2	TCP wrappers	184
7.4.3	Secure shell (ssh)	185
7.4.4	Titan	185
7.4.5	The fix-modes script	185
7.4.6	The SANS scripts	185
7.4.7	The logcheck Perl script	186
7.5	Domino and Notes security basics	186
7.5.1	Notes certificates	186
7.5.2	Certification hierarchies	186
7.5.3	Notes IDs	187
7.5.4	Notes validation and authentication	188
7.6	Protecting a Domino server	188
7.6.1	Protecting access during Domino server setup	188
7.6.2	Setting up basic Domino server security	189
7.6.3	Setting up additional Domino server security	192
7.7	Setting up Domino database security	194
7.7.1	Review database ACLs	195
7.7.2	Consistent ACLs	196
7.8	Anti-virus products for Domino	197
7.8.1	Norton AntiVirus	197

7.8.2	ScanMail	198
7.8.3	Other antivirus products	199
7.9	Summary	199
Chapter 8. Domino Directory services		201
8.1	The Domino Directory	202
8.1.1	Documents in the Domino Directory	203
8.2	The Directory Catalog	204
8.2.1	Server Directory Catalog	204
8.2.2	Mobile Directory Catalog	205
8.2.3	Directory Catalog size	205
8.2.4	Setting up a Directory Catalog	205
8.3	Directory Assistance	211
8.3.1	Setting up Directory Assistance	212
8.4	Extended Directory Catalog	215
8.4.1	How to setup Extended Directory Catalog	216
8.5	Domino LDAP service	218
8.5.1	What is Domino LDAP service	218
8.5.2	Setting up Domino LDAP service	219
8.5.3	Starting and stopping the LDAP server task	220
8.5.4	Showing LDAP statistics	221
8.5.5	Using the ldapsearch utility to search LDAP directories	223
8.5.6	Exporting Domino Directory information	225
8.6	Summary	226
Chapter 9. Domino R5 as a Web server		227
9.1	Solaris Operating System configuration	228
9.1.1	Basic recommendation	228
9.1.2	Network tuning	229
9.2	Domino Web server configuration	230
9.2.1	Settings on a Domino Web server	230
9.2.2	Starting, stopping, and refreshing the Domino Web server	233
9.3	Security	233
9.3.1	Internet certificates	233
9.3.2	Browsing Domino databases via the Internet	234
9.3.3	Domino banner	234
9.3.4	Session authentication	234
9.3.5	Domino Web Realms	237
9.3.6	Domino File Protection	238
9.4	Performance	241
9.4.1	HTTP threads	241
9.4.2	Setting HTTP timeouts	242
9.4.3	Asynchronized Web agents	243

9.4.4	Web statistics	244
9.4.5	Web stress tools	244
9.5	Troubleshooting	245
9.5.1	HTTP does not respond	245
9.5.2	Using the tell command	246
9.5.3	Bindsock issue	247
9.5.4	HTTP thread debugging	247
9.5.5	Memory leaks	249
9.6	Domino R5 console tell commands	249
9.7	Virtual servers and host	250
9.7.1	Network setup	251
9.7.2	Create virtual server or host	252
9.7.3	Create URL mapping and redirection	256
9.8	The Internet Cluster Manager (ICM)	258
9.8.1	Configuration	258
9.8.2	Statistics	262
9.8.3	Troubleshooting	263
9.9	Domino and Java	263
9.9.1	Using a different JVM	263
9.9.2	Java servlets	265
9.10	Domino log and analysis tools	266
9.10.1	Domino Web log	266
9.10.2	Text file analysis	268
9.10.3	Domino Log database analysis	268
9.11	Summary	269
Chapter 10.	Enterprise integration	271
10.1	Domino Enterprise Connection Services	272
10.1.1	Installation	272
10.1.2	Running DECS	273
10.2	Lotus Enterprise Integrator	273
10.2.1	Installation	274
10.2.2	Running LEI	275
10.3	ODBC drivers	276
10.3.1	Configuration	276
10.4	Troubleshooting	278
10.4.1	Test the Oracle environment	278
10.4.2	Using the LCTEST tool	278
10.4.3	Using the CONTEST tool	279
10.4.4	Checking the shared library	279
10.4.5	leiclean	280
10.5	Summary	280

Chapter 11. Backup strategy for Domino R5 on Solaris	281
11.1 Backup strategy	282
11.1.1 Using Solaris backup utilities	283
11.1.2 Backup device options	283
11.2 Backup management	284
11.2.1 Backup versus replication	285
11.2.2 Backup cycles	285
11.2.3 Incremental backups versus full backups	286
11.2.4 Backup using Lotus C API for Domino R5	288
11.2.5 Considerations for backup software	289
11.3 Vendor solutions	289
11.3.1 Legato NetWorker Module for Lotus	290
11.3.2 Tivoli Data Protection for Lotus Domino	294
11.3.3 VERITAS NetBackup 3.4	298
11.4 Planning for successful backups	302
11.4.1 Sample Schedule	303
11.5 Summary	304
Chapter 12. Diagnostics and troubleshooting	305
12.1 Standard procedures	306
12.1.1 Crash	306
12.1.2 Hang	307
12.1.3 Performance problems	308
12.1.4 Packaging the files for support	309
12.1.5 Transferring files to support	309
12.1.6 Common mistakes seen by support	310
12.2 Crash, hang, and performance problem details	310
12.2.1 What is a crash?	310
12.2.2 What is a hang?	311
12.2.3 Poor performance	312
12.3 The NSD tool	312
12.3.1 Running NSD	312
12.3.2 NSD explained	314
12.4 ANSD tool	321
12.4.1 How to use the tool	321
12.4.2 Stack of the crashing thread	322
12.5 The memcheck tool	323
12.5.1 NSD and memcheck	324
12.5.2 Run memcheck manually	324
12.5.3 Open databases	325
12.5.4 Memcheck.dump file	325
12.5.5 Memory dump commands	326
12.6 Core file	326

12.6.1	NSD and the core file	328
12.6.2	The gcore command	328
12.6.3	How to read a core file	329
12.6.4	Killprocess	331
12.7	Troubleshooting	332
12.7.1	Crashes	332
12.7.2	Fault recovery	334
12.7.3	Planning your startup and shutdown scripts	335
12.8	Using the debug_XXX variables	338
12.8.1	How to use	338
12.8.2	Performance issues	338
12.8.3	Typical debug scenario	340
12.9	How to prevent a crash.	341
12.9.1	Maintenance policies.	341
12.9.2	Using NSD	342
12.9.3	Avoid full-text indexes for names and log databases	343
12.9.4	Collecting statistics	343
12.9.5	Show transaction command	345
12.9.6	File descriptor issue	345
12.10	Solaris tools	346
12.10.1	The truss utility	346
12.10.2	The sotruss utility	347
12.10.3	The snoop tool	348
12.11	Troubleshooting tips	349
12.11.1	The server does not start	349
12.11.2	Server crashes or hangs immediately after startup.	351
12.11.3	Semaphore timeouts	352
12.11.4	How to set up semaphore debug	353
Chapter 13. Lotus iNotes Web Access		355
13.1	Lotus iNotes Web Access.	356
13.1.1	High-level overview.	356
13.1.2	Design goals	357
13.2	Installation	359
13.2.1	To install Domino server	359
13.2.2	iNotes Web Access configuration	360
13.2.3	To create user accounts for use with iNotes Web Access	361
13.2.4	To fully enable iNotes Web Access on the browser client	362
Appendix A. Using Notes C API to make your own backup tool		363
Appendix B. Creating a UNIX partition		367
Appendix C. Installing Domino using domsetup		377

Appendix D. Domino and syslog	379
Appendix E. Example TCP port Notes.ini settings	385
Basic Notes.ini settings	386
Standard server - single or multiple IP addresses	386
Multiple IP addresses on the server system	386
Multi-homed server	388
R5.0.x Internet TCP settings	389
Multi-homed server port settings with Internet services	389
Partitioned server - single NIC	391
Partitioned servers - dual NIC	391
Partitioned servers - single NIC and loopback	392
Appendix F. Example script to start and shut down a Domino server	395
Appendix G. Domino IP resolve process	409
Leveraging the Notes name services	411
Leveraging a common secondary Notes name server	413
Using Secondary name servers to back up the user's home server	415
Using a pass-through server	416
Conclusions	417
Appendix H. Additional material	419
Locating the Web material	419
Using the Web material	419
How to use the Web material	420
Related publications	421
IBM Redbooks	421
Other Lotus-related ITSO publications	422
Redbooks on CD-ROMs	423
Also of interest	424
Special notices	425
Index	429

Preface

This IBM Redbook tells you how to run Lotus Domino R5.0.8 on the Sun Solaris 8 Operating Environment. (It contains information that has been revised and updated from the previous edition, which addresses Domino R5.0.2a and Solaris 7.) While the Lotus Domino server is platform-independent, each platform it runs on requires some additional platform-specific knowledge and configuration in order to ensure it operates efficiently and at maximum capability.

The primary focus is to explain the installation, configuration, and performance tuning of Domino R5 in this environment. We take you through all the steps required to run a Domino R5 server on Solaris 8, from choosing the right hardware, installing Solaris and Domino, tuning the OS and the Domino server and performing administrative tasks, through to problem determination and troubleshooting.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Center at Lotus in Cambridge, Massachusetts, USA.

David Morrison is an International Technical Support Specialist for Lotus Notes and Domino at the International Technical Support Organization Center at Lotus Development, Cambridge, Massachusetts. He manages projects whose objective it is to produce redbooks on all areas of Lotus technology. Before joining the ITSO in 1999, he was a senior Lotus Notes consultant working for IBM e-business services in the United Kingdom.

This IBM Redbook was updated for Solaris 8 and Domino 5.08 by the following two authors:

Uwe Wiest previously worked for Lotus Development as a Sales Engineer. He joined Sun Microsystems Inc in 1999 as a System Engineer and today works in the Technology & Product Consulting Department. He works on Lotus Domino projects all over Europe and is responsible for the Sun/Lotus solution partnership in Germany.

Rudolf Utzschneider works as a Senior Consultant for Lotus Professional Services based in Stuttgart, Germany. He is involved in enterprise deployment projects, doing infrastructure planning for complex environments for large client organizations in different industrial sectors. He has been responsible for implementing Domino on Solaris at several client organizations in Central Europe. He also is involved in Domino security projects, performance testing, and infrastructure reviews.

The first edition of this book was prepared by the following authors:

Eric Dolce is a Technical UNIX Support Analyst working for the Lotus EMEA Support Center in Paris. Graduated in Mathematics at the State University of Bologna, he has extensive experience with planning, tuning, and troubleshooting Lotus Domino in all the supported UNIX platforms. He has developed several tools using the Lotus Notes C/C++ API toolkit for UNIX. Before joining Lotus in 1997 he spent 10 years as a software engineer developing code in C/C++ on UNIX platforms for an Italian company of the Olivetti Group. He started to work on UNIX in 1983 with a 3B2 ATT System and has been hooked ever since.

Chris O'Donnell worked for Lotus customer support specializing in UNIX.

Carmen Spieker works as a Consultant for Lotus Professional Services based in Aachen, Germany. While studying Computer Science she was on a work study program where she worked with a Helpdesk solution based on Lotus Notes. Her fascination for Lotus Notes convinced her to work with this product after she had completed her studies. She worked as a Notes Administrator for a reinsurance company before she joined Lotus in 1998. At Lotus she specialized in large scale Enterprise Deployment. Previous project engagements include the setup of a Domino/390 pilot environment for the German army and a worldwide infrastructure for a parcel service on HP-UX. Her last project was the implementation of a Domino R5 based Web-Portal on clustered Sun Solaris machines including Lotus Sametime and other companion products.

Valerie Walker is a Lotus Domino Administrator and UNIX Administrator at the U. S. Geological Survey located in Reston, VA. She has worked with many government agencies specializing in Enterprise Deployment. She has been a UNIX Administrator since 1985 and has worked with Lotus products since the 1980s, first cc:Mail then Notes.

A number of people have provided support and guidance. The team would like to thank the following people:

- ▶ Eddy Bell, Maher Sammer from Iris Development
- ▶ Chris Zaremba, TDP Client Development, Endicott, NY
- ▶ Marc Riart, LPS Spain

- ▶ Dan Jaffe, Shane Kilmon, Anton Lalkens, Jose Lise, Jean-Marc Carta from Lotus Customer Support
- ▶ Jose Ramon Soriano, LPS Spain
- ▶ Charlotte Brooks, Project Leader, Storage Management ITSO, Almaden
- ▶ Glen Kriekenbeck, Dan Parisi, Jay Halloran, Craig Swain, Klaus Ziegler, Eric Sosman from Sun Microsystems
- ▶ Greg Kelleher, Catherine Stone, Terri Gerber, Marc Luescher, Christian Wiest, Diana Ermini, Micheal McCabe from Lotus Development
- ▶ Oliver Froemel, Thorsten Unger from PRS GmbH
- ▶ Martin Meiler and Reto Schuepbach, Credit Suisse Private Banking
- ▶ Shawn Aquino from VERITAS
- ▶ Cindy Cui and Robert Spurzem from Legato
- ▶ Peter Symonds from Tivoli
- ▶ Juergen_Saamen from Trend Micro
- ▶ Kevin Haley from Symantec

Comments welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

Fax the evaluation form found at the back of this book to the fax number shown on the form.

Use the online evaluation form found at
<http://www.redbooks.ibm.com/>

Send your comments in an Internet note to
redbook@us.ibm.com



Introduction to Domino R5 for Sun Solaris

This chapter includes a basic introduction to Lotus Notes and Domino R5. We present a general overview of Lotus Notes and Domino, some new terminology if you're coming from a Release 4 background, and we describe the different components and platforms that make up Notes and Domino.

1.1 Lotus Domino R5 defined

Lotus Domino R5 is Lotus' next generation of Internet products, which includes the Notes Integrated Internet client and Domino server platform for messaging and collaboration, as well as Internet and intranet applications. The Domino Designer is a Web and intranet application development tool. Domino's integrated platform delivers messaging and collaborative solutions for the Internet. Several key services enable IT professionals to manage and run their messaging and Web application infrastructures easily and efficiently. The new R5 Domino Administrator makes it faster and easier for administrators to manage users, configure systems, and optimize performance, all from an intuitive user interface.

1.2 Lotus Notes R5 defined

Notes R5 is the cornerstone of the Lotus client family. Notes R5 provides state-of-the-art e-mail, calendaring, group scheduling, Web access, and information management, all integrated in an easy-to-use and customizable environment.

Expanding access to the powerful Domino server, Lotus' client family also includes Mobile Notes, which provides wireless and mobile devices with access to Domino applications. Also available is iNotes, which brings messaging, collaboration, and off-line support to browsers and Microsoft Outlook users.

1.2.1 Lotus Domino R5 family of servers and clients

The Domino server family is an integrated messaging and Web application software platform for growing companies that need to improve customer responsiveness and streamline business processes.

The Domino server family is comprised of three core servers:

- ▶ Domino R5 Mail Server
- ▶ Domino R5 Application Server
- ▶ Domino R5 Enterprise Server

The Domino client family is comprised of three core clients and one additional sub-component:

- ▶ Domino Administrator
- ▶ Domino Designer
- ▶ Notes client

- iNotes client for Microsoft Outlook and Web Access. iNotes is a component of the Domino server.

In addition, the Domino server supports many of the open standards available today, such as HTTP, IMAP, POP, NNTP, SMTP, LDAP.

1.2.2 New terminology in Lotus Notes R5

New terminology was introduced in the latest release of Domino. Some of the new terms are identified below.

Directory Catalog

The Directory Catalog is a compressed version of one or more Domino Directories, which improves the speed of name lookups and name resolution for all organizations.

Domino Directory

The Public Address Book is now referred to as the Domino Directory.

Directory Assistance

The Master Address Book is now referred to as the Directory Assistance.

Domino Enterprise Connection Services

Lotus Domino R5 includes Domino Enterprise Connection Services (DECS) for building live links between Domino pages and forms to data from relational databases.

Transactional logging

Domino now allows for 24x7 online server backups and recovery support, to eliminate the need to shut down Domino servers in order to maintain them. A transactional log provides a sequential record of every operation that occurs (sequential writing on a disk is much faster than writing in various places on a disk). Logging helps to ensure complete data integrity for updates and enables you to perform incremental database backups.

Domino Off-Line Services

Domino Off-Line Services™ (DOLS) provides a way for browser users to take Domino Web applications offline. Using a browser, an end user can easily take an application offline, make changes, and synchronize the changes with the online application.

1.2.3 Available platforms

Domino's major strength over other products has been its support for multiple platforms. The following tables summarize details about the various operating system platforms that support Lotus Notes/Domino Release 5.0.8.

Note: Operating system patches, servicepaks, and other updates are not specified in the certification tables that follow. Consult the Patch Requirements and Environment Variables sections of the pertinent Release Notes for system updates that should be used with each certified client or server operating system. Also note that operating system vendors frequently release updates. For the most recent information regarding updates, see the Lotus Knowledge Base online at <http://www.support.lotus.com> or contact your local Lotus Support representative. RAM requirements include the minimum required amounts specified by the operating system vendor.

Table 1-1 Support for Domino server (part one)

Feature	Solaris ⁵	AIX	HP-UX
Certified operating system versions	Solaris 8. (see “Solaris patch requirements” release note for patch information)	AIX 4.3.1 (see “AIX patch requirements” release note for patch information)	HP-UX 11.0 (see “HP-UX patch requirements” release note for patch information)
Processors supported	Intel, SPARC	PowerPC, POWER, and POWER2	PA-RISC
SMP support ¹	Yes	Yes	Yes
RAM	64MB minimum 128MB or more recommended	64MB minimum 128MB or more recommended	64MB minimum 128MB or more recommended
Disk space ²	750MB minimum 1GB or more recommended	750MB minimum 1GB or more recommended	750MB minimum 1GB or more recommended
Disk swap space	3 times the physical RAM recommended	3 times the physical RAM recommended	3 times the physical RAM recommended
Monitors supported	Color monitor required	Color monitor required	Color monitor required
Protocols supported			
AppleTalk	No	No	No
Banyan VINES	No	No	No
ISDN ³	No	No	No
NetBIOS/NetBEUI	No	No	No
SNA ³	No	No	No
SPX ⁴	No	Yes	No
SPX II	No (Intel) Yes (SPARC)	Yes	No
TCP/IP	Yes	Yes	Yes
X.25 ³	No	No	No
X.PC	Yes	Yes	Yes

Table 1-2 Support for Domino server (part two)

Feature	Linux ⁸	OS/2	Windows NT
Certified operating systems and (for Linux only) supported operating systems	Certified: Red Hat 6.0 Intel x86 Supported: Caldera 2.2 Intel x86 (see the "Linux patch requirements" and "Linux settings" release notes for additional important information)	OS/2 Warp Server 4 - Entry; Warp Server 4 - Advanced; Warp Server 4 (with SMP Feature) (see "OS/2 operating system fixpacks" release note for patch information)	Windows NT Server 4.0; Windows NT Workstation 4.0 (see "Windows Service Packs" release note for SP information)
Processors supported	Intel x86	Intel	Intel Pentium, Alpha
SMP support ¹	Yes	Yes	Yes
RAM	64MB minimum 128MB or more recommended	48MB minimum 64MB or more recommended	48MB minimum 96MB or more recommended
Disk space ²	750MB minimum 1GB or more recommended	750MB minimum 1GB or more recommended	750MB minimum 1GB or more recommended
Disk swap space	3 times the physical RAM installed	16MB minimum	64MB
Monitors supported	Color monitor required	Color monitor required	Color monitor required
Protocols supported			
AppleTalk	No	No	Yes (with Service Pack 3 or higher)
Banyan VINES	No	No	Yes (Intel Pentium) No (Alpha or SMP)
ISDN ³	No	No	Yes
NetBIOS/NetBEUI ⁷	No	Yes	Yes
SNA ³	No	No	Yes
SPX ⁴	No	No	Yes
SPX II	No	No	Yes
TCP/IP	Yes	Yes	Yes
X.25 ³	No	No	Yes
X.PC	Yes	Yes	Yes

Important: Not meeting minimum recommended patch requirements on the operating system that underlies Domino Server can cause serious system instability. Be sure to read the patch requirement documents in the Release Notes regarding the platforms you use. These platform-specific documents can be found in the Release Notes “Things you need to know” chapter.

Table 1-3 Client support: Notes, Domino Designer, and Domino Administrator

Feature	Windows 95/98	Macintosh	Windows NT
Certified operating system versions	Windows 95; Windows 98 (see “Windows Service Packs” release note for SP information)	Mac OS 9	Windows NT Workstation 4.0 (see “Windows Service Packs” release note for SP information)
Processors supported	Intel Pentium	PowerPC	Intel Pentium
RAM	8MB minimum 32MB or more recommended	32MB physical, 64MB virtual minimum 64MB physical, 80MB virtual recommended	16MB minimum 32MB or more recommended
Disk space The minimum amounts are the disk space required for installing default files. More disk space is required if databases are replicated locally or copied locally.	Notes client: 69MB minimum 112MB or more recommended Designer client: 70MB minimum 236MB or more recommended Administrator client: 78 MB minimum 182 MB or more recommended	75MB minimum 100MB or more recommended (standard client) 75MB minimum 150MB or more recommended (designer client)	Notes client: 69MB minimum 112MB or more recommended Designer client: 70MB minimum 236MB or more recommended Administrator client: 78MB minimum 182MB or more recommended
Monitors supported	Color monitor required	Color monitor required, 256 colors or greater.	Color monitor required
Protocols supported			
AppleTalk	No	Yes	No
Banyan VINES	Yes	No	Yes
ISDN ³	Yes	No	Yes
NetBIOS/NetBEUI	Yes	No	Yes
SNA ³	No	No	No

Feature	Windows 95/98	Macintosh	Windows NT
SPX	Yes	No	Yes
SPX II	No	No	Yes
TCP/IP	Yes	Yes	Yes
X.25 ³	No	No	No
X.PC	Yes	Yes	Yes

Table notes:

1. SMP (Symmetrical Multiprocessing) support is for SMP-enabled versions of listed operating systems. For details on whether a version of an operating system supports SMP, check with the operating system vendor or with your Lotus representative.
2. Disk space requirements include estimated free disk space amounts for a functioning Domino system (that is, one or more mail databases and applications). The actual disk space needed to install the Domino files is lower than the minimum and recommended values.
3. Notes WAN Drivers (Connect for X.25, Connect for SNA, and Connect for CAPI ISDN) are available for download from <http://www.lotus.com>.
4. Domino Clusters and Partitioned server configurations do not support the IPX/SPX protocol. At this time, Lotus does not plan to provide IPX/SPX network support for future releases of these features.
5. Starting with Domino release 5.0.4 Solaris 8 is the certified platform when running Domino in 64-bit mode. Please refer to the "Solaris for Intel and SPARC - support and certification" Release Note for further details.
6. The Domino cache directory must reside on an HPFS file system. The cache directory is specified in the Notes Server document under the HTTP Server section.
7. Notes SPX and NetBIOS port driver (Novell NetBIOS) is not certified or supported on an OS/2 Warp server platform.
8. A certified Domino Server was first made available for Linux in the R5.0.2 Domino Server release. For Domino on Linux, only English locales are supported.



Sizing and configuration of Solaris

In this chapter we discuss the decisions that must be made in your organization to select the hardware that is necessary for the Domino server on a UNIX Solaris platform. Proper sizing is the first step to a reliable environment that can handle your Domino applications and provide a stable platform that provides high availability and scalability.

After you have selected the memory, disk space, and CPU, you must configure your system correctly to be able to work optimally with the Solaris OS and the Domino server. We describe in detail the things that should be done to the system prior to installing the Domino binary software.

If you are currently running Domino on existing NT platforms, and would like to move to a Solaris OS, we explain the things that should be done to move your existing servers to Solaris.

We begin this chapter with a brief introduction to the Solaris 8 operating system.

2.1 What's new in Solaris 8

There are many new and improved features offered in the Solaris 8 environment. We review some of the features in this section; for more detailed information on the changes and enhancements, see the white paper available at:

<http://www.sun.com/solaris/ds/ds-sol8oe>

Note: Lotus and Sun recommend the use of Solaris 8 with Domino R5.0.4 and later.

2.1.1 Solaris 8 Admin Pack

The Solaris Easy Access Server product is an extension to the Solaris Operating Environment that provides increased interoperability, security, and workgroup functionality. This previously separate product is now included with Solaris 8. The Easy Access server products can now be downloaded from a Sun Download Center. These services are tools that can be used by the system administrator. The Solaris 8 Admin Pack contains much of the same functionality as found in Solaris Easy Access Server, and is available as a free download from the Solaris System Administrator Portal at

<http://www.sun.com/bigadmin/content/adminPack>

The Solaris System Administrator Portal is the primary resource for content and value-add services relating to Solaris system and network administration.

For more details visit: <http://sunsolve.sun.com/>

2.1.2 Diagnostic and configuration tools

Keeping your system updated with the latest patches is something every administrator should do on a periodic basis. These are some of the ways in which you can do this on a Solaris system.

- ▶ **Hot Patching for Diagnostics**
This feature allows you to apply patches to most areas of the system without rebooting. This greatly reduces the downtime to diagnose, test, analyze, and correct operating environment problems.
- ▶ The **proc** system tool has been enhanced for easier analysis of application crash (core) files. The **proc** tools are utilities that can manipulate features of the /proc file system.
- ▶ In addition to the proc tools, the configuration command **coreadm** has been added to greatly improve system-wide management of application crash (core) files.

- ▶ A new command for system configuration, **devfsadm**, provides an improved way to manage the special device files in the /dev and the /devices directories.
- ▶ The system boot and error message formats now provide a numeric identifier, module name, and time stamp for messages generated by the syslog(1M) logging facility. In addition, messages that were previously lost after a system panic and reboot are now saved.
- ▶ System events and messages that are written to the local system console can now be redirected to a network-connected remote console.

2.1.3 Installation and management

The following are some of the new installation and management features of Solaris 8.

- ▶ **Solaris Web Start**
Installs operating environment and other software using a Web browser interface.
- ▶ **WBEM**
An implementation of Web-Based Enterprise Management on Solaris 8. WBEM is used to ease the development of applications that manage Solaris software systems and administer the Solaris operating environment.
- ▶ Perl 5 is included in the Solaris 8 operating environment.
- ▶ Support has been added for the Lightweight Directory Access Protocol (LDAP). It has been included in the Solaris naming service switch (nsswitch) to search in the LDAP directory for names in addition to NIS and NIS+.
- ▶ Two new process ID commands were added:
 - **pgrep** - Looks at the active processes on the system and displays the process IDs whose attributes match the specified criteria on the command line.
 - **kill** - Works similarly to the pgrep command, except that each matching process ID is signaled by kill instead of process ID. See documentation for a further explanation.

2.2 Sizing requirements for a Domino mail server

In this section we provide a basic sizing guideline for configuring a Solaris system for a common Domino R5 mail server. The configuration is based on documentation provided by Sun and Lotus from real-world feedback, and Sun's internal sizing guidelines, as well as in-house testing using tools such as NotesBench™.

Important: Please note that this section details sizing guidelines for a standard production Domino mail server being accessed primarily by Notes clients. It does not cater to any other type of Domino server configuration.

This information is provided as a guide to give you an *idea* of the platform and size of your Solaris system. For a more accurate sizing for your particular business, you may need to contact a Lotus and/or Sun Microsystems consultant or use further documentation.

The first thing that needs to be done to size your Solaris server to support Domino is to gather information related to what you aim to do with the server. This information can be gathered from your existing Domino server environment or any other mail server environment. If you presently do not have e-mail, estimates can be made from your total employee population.

- Total number of registered users (Domino Mail server)

This is the total number of users registered in Domino. If you do not have a Domino server installed, determine this number from your existing mail server or estimate it based on how many of your current total employees will have access to e-mail.

- Peak concurrent usage percentage (Domino Mail server)

The peak concurrent usage percentage can be determined from the “show server status” command on the console in the Administration client. If you do not have Domino installed, determine the percentage of users that are using your current e-mail system at one time. If you do not have any existing e-mail system, you will need to estimate a percentage of use. If it is unknown, begin at 50 to 60 percent as a starting point.

- Average mail file size (Domino Mail server)

Determine the average current mail size of your Domino implementation, using the Administration client. If you do not presently have Domino installed, determine the average disk space used by your users for storing e-mail (include attachments). If you do not presently have an e-mail system installed, consider the average Domino mail file size to be 50 MB.

- ▶ Description of Domino usage (e-mail, calendaring and scheduling, Web server, and so forth)

Determine what you are primarily using your Domino servers for. If you are not presently using Domino, estimate the primary future use.
- ▶ Type of clients (Notes, IMAP4, SMTP/POP, browsers, and so forth)

Determine the e-mail format that will be used. Notes mail is a client-based email/calendaring/scheduling application installed on a workstation to access a personal database located on a Domino server. IMAP4 is used primarily for client/server e-mail and MIME processing. POP3 is also used for client/server e-mail, using a workstation to retrieve mail that a server is holding for it. Examples of POP3 clients are Netscape and Eudora. Webmail is e-mail accessed through an Internet browser such as Netscape or Internet Explorer.
- ▶ Network topology

Identify the type of Networking Services that you will use for Domino. The supported Networking Services are NetWare and TCP/IP.
- ▶ Size of other Notes/Domino databases (discussion, applications, Web servers, and so forth)

The determination of CPU, disk space, and memory required can be calculated by the results of this information.

Note: Concurrent usage is defined as the total number of users logged on and accessing the system at one time. If the primary use of Domino is for e-mail, typical peak concurrent usage is often in the 30 to 40 percent range of your total registered users. It is recommended that you add another 5 to 10 percent to this estimate to ensure an adequate configuration. If you cannot determine your expected concurrent usage, begin with a 50 percent concurrent user load.

2.2.1 CPU

The number of concurrent users a CPU will support varies by the type of workloads the users generate. Table 2-1 is a guideline you can use to estimate the number of concurrent Notes clients that can be supported by two common types of UltraSPARC CPUs being sold by Sun at the time of this writing.

Table 2-1 Guidelines for the number of peak concurrent users supported per CPU

	~30 MB mail files		~75 MB mail files		~200 MB mail files	
CPU	Mail only	Mail & Calender	Mail only	Mail & Calender	Mail only	Mail & Calender
750 MHz	~1260 users	~1010 users	~860 users	~690 users	~650 users	~520 users
400 Mhz	~840 users	~670 users	~570 users	~460 users	~430 users	~320 users

Important: The ~ character denotes an approximation. The figures in Table 2-1 are only to be used for guidelines. Contact Lotus and Sun for more accurate information on CPU sizing.

For more information on sizing Domino partitions see 2.4.5, “Domino partitioning” on page 22.

2.2.2 Memory

As with CPU, memory utilization will vary depending on the workload being supported. Use the following to estimate the memory requirements for a generic Domino R5 mail server being used by Notes clients after you have determined the number of CPUs required as described in the previous section.

- ▶ For each 750 MHz CPU - 2 GB memory is preferred.
- ▶ For each 400 MHz CPU - 1 GB memory is preferred.

2.2.3 Disk space

Next determine the amount of disk space that will be required for the Solaris operating environment, virtual memory management (swap), and the Domino application software.

▶ **Solaris operating environment**

The minimum requirement is 600 MB. Allocating at least 1 GB for Solaris, logs, and so forth is more appropriate.

▶ **Virtual memory management**

The Solaris operating environment requires disk space (swap) to manage the virtual memory subsystem. For Domino servers with less than 1 GB of memory, configure 3 times the amount of physical memory for swap space. For larger servers configure 5 GB (and no more) of swap space per Domino partition.

The number of disk spindles (separate hard drives) is as important as the actual amount of disk space. Multiple spindles allow the I/O to be distributed more efficiently, preventing performance bottlenecks.

▶ **Domino program directory**

The Domino program directory is where the Domino executables and shared program files are stored. Allow a minimum of 300 MB.

▶ **Domino data disk space**

The Domino data structure is divided into four parts: transactional logging

space, Domino data directory, user mail files, and application databases. Each part has its own storage requirements.

- Domino transactional logging directory
Allocate a separate physical disk of 4 GB or greater for each Domino partition. (We discuss partitions later in this chapter.) See also Section 11.1, “Backup strategy” on page 282 for information on how to back up this space.

Important: The transactional log disks should not be used for anything else other than storing the transaction logs.

- Domino data directory
Allocate approximately 200 MB per partition. (We discuss partitions later in this chapter.) The Domino data directory is also used to store all the Domino data files, including the names.nsf database. However, with large Domino directories you may need to allocate substantial space for this directory. 1 to 2 GB is not uncommon in a large organization.
- User mail files
Allocate disk space based on the calculation of number of users multiplied by the average mail size. If several mailboxes far exceed the average user mailbox size, add some additional space to the results of the equation. Mail files can typically range from 75 MB upwards to over 1 GB per mail file for heavy mail users. It is therefore important to consider this when determining the disk space for these files.
- Databases
If you will be implementing additional Notes or application databases such as discussion databases, document libraries, workflow applications, and so forth, add disk space for these also. If the server will be supporting e-mail only, you may not need to factor in these amounts.

Once you’ve calculated the amount of disk space, you need to adjust it for the UFS file system overhead and to provide optimal disk performance. This is calculated as follows: Divide the data space requirements by 0.9 to account for file system overhead and directory structures. Full disk subsystems will often perform poorly, so for optimal performance, divide this result by 0.8.

For example: Assuming there are 3,500 users, each with 75 MB mail files and a 1.5 GB names.nsf, the calculation of data disk requirements would be:

200 MB for the additional Domino files such as templates
1,500 MB Domino Directory (names.nsf)
3,500 users with 75 MB mail files = 262,500 MB

Total file space is $200 + 1,500 + 262,500 = 264,200$ MB
 Add UFS overhead = $264,200 / 0.9 = 293,556$
 Add additional space for optimal performance = $293,556 / 0.8 = 366,945$ MB
 Total disk space required is therefore 367,000 MB or 367 GB

Note: Refer to 2.5.4, “File system layout” on page 36 for more information on disk layout, including RAID.

Table 2-2 presents a summary or hardware requirements for a Domino mail server running under Solaris 8.

Table 2-2 Sample disk requirement by CPU and size

Configuration Size	Solaris OS	Swap Space (Paging) on striped disks	Domino Server Code and Files	Number of Disks Required
Small: 1 CPU 512 RAM	600 MB	3 times RAM = 1536 RAM	750 MB minimum 1 GB recommended	1 disk
Medium: 2 to 6 CPUs 2 to 4 GB RAM	2 GB on 1st disk	Equals RAM on 1st disk - 2GB on 2nd disk	2 GB 2nd disk	2 disks
Large: 4 to 10 CPUs 4 to 8 GB RAM	2 GB on 1st disk 2 GB for Solaris OS logs on 3rd disk	Equals RAM on 1st disk 2 GB on 2nd disk 2 GB on 3rd disk	2 GB on 2nd disk	3 disks

Note: Additional disks are required for the following scenarios:

- For the systems defined in Table 2-2, add one disk for each Domino partition if transaction logging is enabled.
- Disk space for the Domino data directory varies depending on usage.

2.2.4 Network I/O sizing

Studies have shown that typical bandwidth required for an active Notes user ranges from 3.6 kbps/sec to 4.0 kbps/sec. This does not account for the practical network utilization supported by various network topologies (LAN, MAN, WAN).

In a WAN environment, for example, if we have a dedicated T1 WAN connection, and allowing for the overhead of the networking protocol at 50%, we can expect this to support the following:

$$1.5\text{Mbit} * 50\% = 750\text{Kbits/sec}$$

750Kbits/sec / 4Kbits/sec per user = 187 active Notes users.

Note: For WAN connections, the gating factor will be the slowest link in the pathway between the Notes client and the Domino Server.

In a LAN environment, if we have a dedicated 10 Mbit connection, and allowing for the overhead of the networking protocol at 60%, we can expect this to support the following:

10Mbit * 60% = 6000Kbits/sec

6,000Kbits/sec / 4Kbits/sec per user = 1,500 active Notes users.

Note: With 1,500 active Notes users you would saturate this network segment. You may need to investigate other alternatives to increase the network bandwidth to the machine, such as 100 Mbit and/or multiple network cards and segments.

2.3 Solaris patch considerations

For the latest patch levels check the release notes that ship with Domino and can be found on the installation CD in an Adobe Acrobat PDF file. They can also be found in the domino/data/doc/readme.nsf database once Domino has been installed. In addition to the readme file, you can also check on <http://notes.net> for the latest patch recommendation and requirements.

The Domino release notes on Notes.net specify the minimum required Solaris patches. The Domino support organization also supports the use of Solaris updates and patch clusters that include more recent versions of these required patches. It should be up to the administrator to ascertain whether to install these clusters or individual patches.

2.3.1 Checking for available patches

CheckOS is a patch checker utility that can be run manually and displays missing patches for your operating system (OS). If all of the correct patches are installed, you are notified with the message, "The OS appears to have the correct patches." This feature is new in Release 5.0 of the Domino server. See also 3.1.1, "Install and configure the Solaris environment" on page 44 for more information on CheckOS.

The following example installs a patch to a standalone machine:

```
example# patchadd /var/spool/patch/104945-02
```

The following example removes a patch from a standalone system:

```
example# patchrm 104945-02
```

For additional examples please see the appropriate man pages.

Tip: Patch Check is a tool written in Perl that can be downloaded from the addresses that follow. Sun provides a cross-reference file from which this tool can compare which patches on your system are missing or should be updated. Not every patch that this tool shows available should be installed, so carefully read the README file of the patches you want to install.

Note: All patches for the current release of Domino are in the release notes database or the readme.pdf file on the Domino server CD-ROM.

Patches that relate to your Solaris Operating System version should be reviewed periodically. Ideally, security patches should be reviewed regularly and kernel patches should be reviewed for relevance to your environment.

Patches can be obtained from Sun on the following Web sites:.

```
http://access1.sun.com/  
http://sunsolve.sun.com/
```

2.4 Operating system considerations

In the previous section, we discussed the information that you need to gather to configure your Solaris system to run Domino correctly for your environment. Next, we continue this process by discussing the additional requirements you may have for special needs in your business.

The main reason for implementing Domino on a Solaris platform is to take advantage of the reliability and scalability of the operating system (OS). To get the most out of the features of Solaris, you may want to design special configurations that allow you to implement the large-scale configuration that Solaris can provide for the Domino servers.

In this section we describe the services that are available with Solaris that work in conjunction with your Domino R5 implementations. The services are:

- ▶ Disk drive configuration
- ▶ Configuring /tmp

- ▶ Domino partitioning
- ▶ Domino clustering
- ▶ Hardware clustering

After we explain how each of these features is implemented on a Solaris platform, we discuss the OS environment tuning that should be done to utilize the power of the operating system and to ensure that the Domino implementation is correctly aligned with the OS. We only discuss the initial tuning that relates directly to the Solaris OS here; later in this book we discuss tuning specifically for optimal operation of the Domino server.

The areas to be covered in the OS environment tuning are:

- ▶ Disabling default services and processes
- ▶ Entries in the `/etc/system` file

2.4.1 Disk drive configuration

Part of the installation process on a Solaris platform is to select the disk drive that will hold your Solaris OS and to configure this drive. To configure the drive you will need to partition it. Partitioning in the Solaris environment is not the same as partitioning in relation to your Domino server. When a disk is added on a Solaris system you must decide the sizes for the different file systems. A file system is defined as the physical or logical device that holds a collection of files and directories. This might be a hard disk drive or a partition on a disk drive. Figure 2-1 displays a basic UNIX partition setup for the Solaris Operating system.

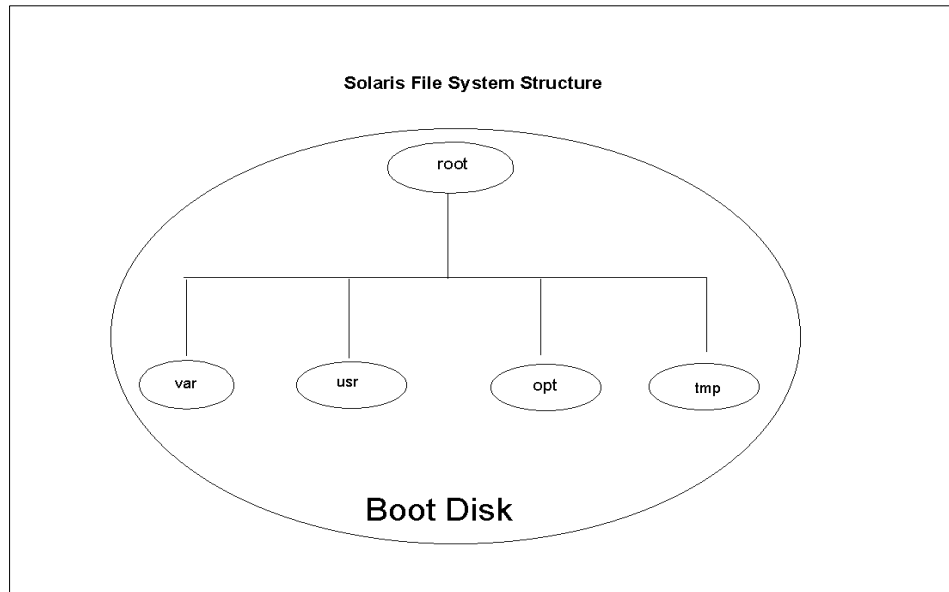


Figure 2-1 Solaris file system structure

This figure shows how a UNIX file system is set when using the primary disk drive and partitioning five sections for the different system directories. For further information on UNIX partitioning and UNIX File Systems, consult the many UNIX books available, as well as various sources on the Internet.

2.4.2 Configuring /tmp (Swap)

Domino on Solaris utilizes mmap() files which are created under /tmp and used in place of System V shared memory. This provides significant speed increases over traditional System V shared memory. In Solaris, the /tmp file system is a memory based file system whose size is determined by physical memory plus the size of all the swap space. To avoid any I/O bottlenecks, refer to the discussion earlier in this chapter regarding disk space, particularly Table 2-2 on page 16. In order to use /tmp effectively, you should partition one portion of each of your disks for /tmp (swap) area.

2.4.3 Configuring default services

On a standard Solaris install there will be a number of daemons running in the background, which you may want to disable for security and performance reasons. It is beyond the scope of this book to address all of these services; refer to one of the many Solaris administration and security books which cover this topic in depth. As these administration references describe, you may wish to

bind these services exclusively to particular IP interfaces. If you decide to disable any Solaris services, make sure you understand the ramifications of the services you disable. For instance, on a Domino application server you might disable FTP service running in the background due to potential security issues; however, this will disable your ability to use the FTP server to transfer files to Sun/Lotus support, for example.

For more information on Domino R5 security, refer to the IBM Redbook “Lotus Notes and Domino R5.0 Security Infrastructure Revealed”, SG24-5341-00.

Two services that ship with Solaris—sendmail and HTTP—should be considered before you install Domino. Domino and Solaris can both provide SMTP services for mail and routing. You will need to disable Solaris sendmail if you want to use Domino’s service. The easiest way to stop this process is to prevent it from starting. To do this, edit the command files in the directory `/etc/rc2.d`. If you change the file names from a capital letter to a small letter, the init daemon will not recognize the scripts as startup scripts.

Example: `mv S88sendmail to DISABLED.S88sendmail`

HTTP Web and application services are often installed on Solaris servers. Domino HTTP services and other HTTP services can coexist if they are configured to avoid IP address/port conflicts. For example, you can set Apache to listen on port 8080 and Domino to listen on port 80.

2.4.4 Edit Solaris system configuration (`/etc/system`)

Solaris kernel parameters are set in the `/etc/system` text file. Software applications may require that you set parameters here to override the Solaris default settings in order for the application to function. You may also decide to set parameters here to tune the system to optimize performance.

As of the time of this writing, with Solaris 8 and Domino R5.0.8, the only parameter you need to set in `/etc/system` is:

```
set rlim_fd_max=65536
```

This parameter raises the maximum number of file descriptors a single process can have from the default of 1,024 to 65,536.

You can set additional parameters in `/etc/system` to tune your system; this is discussed in more detail in Chapter 4, “Tuning Domino Server on Solaris” on page 91.

The following is a sample `/etc/system` file with just the `rlim_fd_max` parameter set:

```
*ident "@(#)system 1.18 97/06/27 SMI" /* SVR4 1.5 */
```

```
*  
* SYSTEM SPECIFICATION FILE  
* Lotus Domino settings below:  
set rlim_fd_max=65536
```

Tip: If you are upgrading from Release 4 of Lotus Notes to Domino R5, start with just the `rlim_fd_max=65536` setting in the `/etc/system` file. Refer to Chapter 4 for more information to determine if additional tuning settings are appropriate.

Note: Changes in the `/etc/system` file require a REBOOT of your system.

2.4.5 Domino partitioning

Domino partitioning is when multiple instances of the Domino server run on the same machine. Currently in R5 there is no theoretical limit on the number of partitions you can run, given infinite hardware resources. In practical terms, however, you will find that too many partitions on one physical box will quickly put a strain on the hardware. The number of partitions that is right for any given machine depends on the hardware available and the expected load on each partition. The type and frequency of client access to the server will also need to be taken into account. You can make some rough estimates using guidelines, but even the best planning can leave your server overburdened, forcing you to scale back the number of clients accessing the partitions in question or the entire machine.

As a general rule, we recommend:

- ▶ Limiting the number of partitions to the number of CPUs on your machine. Also keep in mind that the load on each Domino partition depends on the workload and tasks it is performing. As a rule of thumb a Domino partition can handle approximately 2,000 concurrent typical Notes client users. Overloading the system with additional users may affect the performance or stability of the server.
- ▶ Each partition will require its own data directory. Where possible each data directory should be located on a separate physical device.
- ▶ There is no limit on the number of partitions, but the sample startup script in this book only handles up to 9 partitioned servers (notes 1 to 9).

2.4.6 Domino clustering

Clustering provides high availability and scalability to the Domino environment. With clustering you can have multiple replica copies of databases on up to six servers. Automatic load balancing within the cluster allows Domino to redirect users to servers with the least amount of load. By adjusting the threshold on each server, the administrator can efficiently govern the maximum load against each machine in the cluster. As the demand on the cluster grows, more servers can be easily added, offering greater expandability.

The process used to cluster Domino servers is to set up one Domino partition on a server to replicate (copy) the databases to another Domino server partition on a separate server. Using the agents that manage the cluster, the server can be configured to have the users connect to the cluster portion of the server in the event that the primary Domino partition goes down. This feature can also be set to handle load-balancing. In the circumstance that there are too many users accessing the server at one time and performance is slowing down, users can be redirected to the cluster server to distribute the load and balance the utilization of the servers.

The steps for setting up partitions and clusters are in the next chapter.

Cluster Manager

The Cluster Manager resides on each cluster member and is responsible for exchanging messages with other cluster members (probes) to determine who is available in the cluster and their current capacity. The Cluster Manager decides where to send a connection request based on this data and reports server status to other cluster members.

Cluster Administration

Cluster Administration creates the database `clbdbir.nsf`, which defines the databases contained within the cluster. The other components of clustering use this database to perform their functions. Cluster Replicator will use this database to decide which databases to replicate.

Cluster Replicator

Cluster Replicator handles the replication of the databases within the cluster. Replication in clustering is event-driven. When a change is made to a database, the Cluster Replicator immediately propagates the change to the other cluster members.

2.4.7 Hardware clustering

Solaris supports a number of hardware cluster solutions, including Sun Clusters and Veritas Cluster Manager. Currently, there are no vendor supplied Domino modules for these clustering services. Administrators have used the scripting capabilities of these products to create custom solutions.

2.4.8 Network redirection

Certain services are available which can redirect network traffic to multiple Domino servers for load balancing and availability. Not all services can be redirected because of the session type (stateful and stateless). A common example is to load balance and achieve high availability of an HTTP Web server by using an HTTP redirector (IP sprayer). A redirector handles load balancing, failover, and sessions for the Web server.

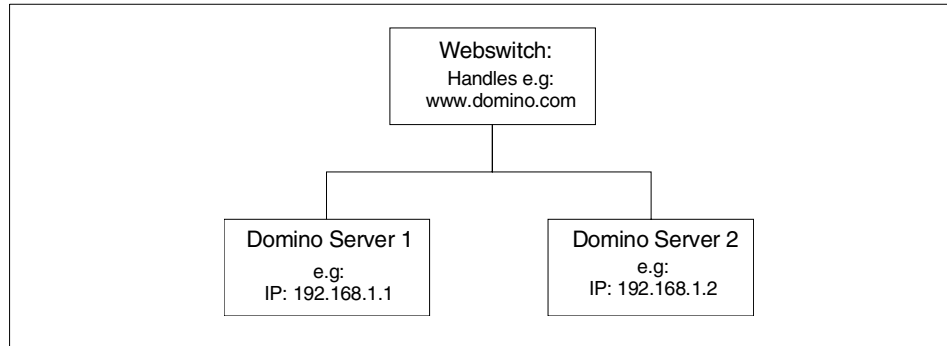


Figure 2-2 Load balancing with a HTTP redirector (Webswitch)

The users access the website `www.domino.com` and will be automatically transferred to the server Domino1 or Domino 2 by the HTTP redirector, depending which of the servers is available. If both are available, the webserver with the lower number of sessions will be chosen by the HTTP redirector. In case of a server failure the session will be automatically transferred to the other server.

2.5 Configuration checklist

In this section we describe how to set up your Solaris system in preparation for the installation of Domino software. With the information provided you should be able to install your Solaris OS and get your system up and running. For step-by-step installation instructions for Solaris 8, refer to the Solaris product documentation. Before you install your Domino software, there are some changes that you must make to your Solaris system.

2.5.1 Creating users and groups

Before installing Domino software, you should first create a Solaris group for the Domino server. You should also create a Solaris account for each server to be installed. If you are installing multiple Domino partitions on a server, we recommend that you create a separate Solaris account for each partition to make each partition easier to administer, using the same common group. In addition, if you need to create a Solaris account for yourself for administrative purposes, follow the same procedures listed in this section.

In the following steps we will use **admintool** to create a new group and account called “notes.” If you have multiple partitions on the server you could create accounts “notes1,” “notes2,” “notes3,” and so forth, for each partition of Domino you are installing. Assign the group “notes” in the group field.

Note: The example script provided in Appendix F, “Example script to start and shut down a Domino server” on page 395 assumes that you have created Solaris accounts named notes1-notes9. If you choose a different naming convention, then this script will need to be modified.

The name “notes” is used here, but is not a required name for the account. It could be considered a security risk to require that the account name only be “notes,” as it would make for easier access to hack into your system. In the script provided with this book you should use the UNIX user names *notes1* to *notes9* and the UNIX group *notes*. You may want to use a naming convention that uses a theme that combines the system name with the Domino server name. For example create an account named SaturnNotes1.

Note: Log on as the root Solaris Account or **su** to that account. (**su** is a UNIX command to switch user to a UNIX ID. You must have root user access level permissions to add users.)

Use the following steps to start the **admintool** utility and create a group: At the UNIX prompt type **Admintool** and press Enter. The **Admintool** application will appear, as shown in Figure 2-3.

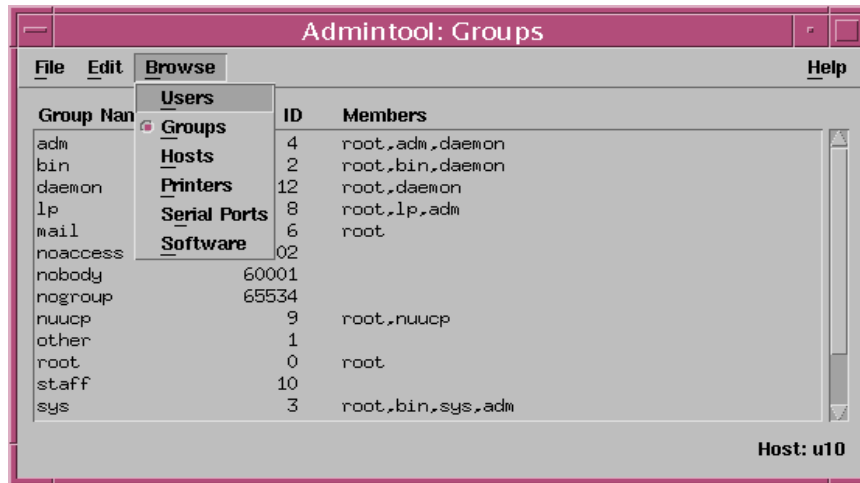


Figure 2-3 Adding groups with the Admintool program

1. From the Admintool menu, select **Browse -> Group -> Edit -> Add**. The Add Group dialog box shown in Figure 2-4 on page 26 will be displayed.

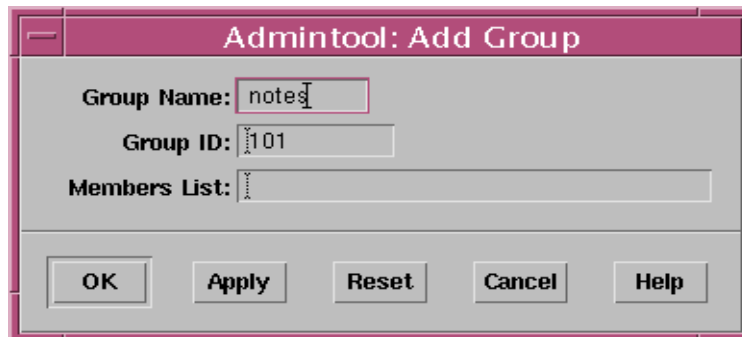


Figure 2-4 Adding a new Notes group

2. Enter a new group name for the Solaris group for Domino and accept the group ID provided.
 3. Click OK.
- Important:** Make note of the group ID number for reference when creating the Notes user ID.

Use the following steps to create your Solaris account ID for Notes:

1. From the Admintool menu, select **Browse -> User -> Edit -> Add**. The Add User dialog box will be displayed.

Admintool: Add User

USER IDENTITY

User Name: notes1

User ID: 1001

Primary Group: notes1

Secondary Groups:

Comment: Domino user notes1

Login Shell: Bourne ☐ /bin/sh

ACCOUNT SECURITY

Password: Normal Password... ☐

Min Change: 0 days

Max Change: 0 days

Max Inactive: 0 days

Expiration Date: None ☐ None ☐ None ☐
(dd/mm/yy)

Warning: 0 days

HOME DIRECTORY

Create Home Dir: ☒

Path: /lotus/notes1

OK Apply Reset Cancel Help

Figure 2-5 Adding a new user with the Admintool program

Note: Which login shell you select is a personal preference. We have selected the bourne shell in our example, but you may wish to use another. If you select a different shell the configuration and environment filenames will be different than those listed in the following sections.

Important: If you want to use the script delivered in this book you have to use the bourne shell.

2. Enter your Solaris account ID for Domino, and leave the default user ID number.
3. In the Primary Group field, type the Solaris group ID number assigned when you previously created the Notes group or enter the Solaris group name you created.
4. Enter in the comment field a brief description of the account you are creating, for example, Domino Account id for first partition
5. Select the login shell you will use. Most people prefer the C shell. The default is the Bourne shell. However, there is a growing interest in the Korn shell.

Note: For further information on UNIX shell variables consult your UNIX documentation.

6. In the next section you have options for the password setting. Click the push button next to Password; a pull-down list appears. Select one of the following options:
 - Normal Password
This allows you to set the password now. If this option is selected you will receive another window, in which to enter the password and then verify it. When you have done this, click OK.



Figure 2-6 Setting a users password

7. Do not modify the password retention fields unless you have a specific reason to change them and make the login passwords temporary.
8. The last section of this dialog box, labeled HOME DIRECTORY, is used to create the home directory for the Solaris Account ID for Domino that you are creating. The home directory will contain a number of Solaris-related files; therefore, we recommend the Notes data directory be a subdirectory of the home directory. For example, you might use /lotus/notes1 for the first account home and data directory.

Tip: If you have multiple Domino partitions for your server, each Domino partition requires a Solaris Account ID. Each Solaris account ID will have its own Solaris home directory and its own Domino data directory.

Example: We have established two partitions on our system for the Domino servers.

Table 2-3 Adding Solaris accounts for Domino partitions example

Domino Partition	Solaris Account	Solaris Home Directory	Domino Data Directory
1	notes1	/lotus/notes1	/lotus/notes1/notesdata
2	notes2	/lotus/notes2	/lotus/notes2/notesdata

Admintool: Add User

USER IDENTITY

User Name:

notes2

User ID:

1002

Primary Group:

notes

Secondary Groups:

Comment:

Domino user notes2

Login Shell:

Bourne

/bin/sh

ACCOUNT SECURITY

Password:

Normal Password...

Min Change:

days

Max Change:

days

Max Inactive:

days

Expiration Date:

None

None

None

(dd/mm/yy)

Warning:

days

HOME DIRECTORY

Create Home Dir:

Path:

/lotus/notes2

OK

Apply

Reset

Cancel

Help

Figure 2-7 Adding a second user for Domino partitioning

9. Click OK.

2.5.2 UNIX environment for Domino user ID

After you have created the Notes group and Solaris account ID, you will need to configure the environment for the Solaris account user ID used for Domino.

The UNIX environment defines the way your Solaris account ID will work when you log on to the Solaris system. It defines which X-Windows you will use, what executables you have access to by default, if you will receive e-mail, and so forth. There are many things that will not be utilized on a Solaris system that are *only* used by the Domino server. The Domino server-specific settings are described in this section.

- ▶ Edit the configuration file to have a PATH statement that includes the binary directory for the Domino software. The following example is for the Bourne shell in the .profile file:

```
# @(#)local.profile
PATH=/usr/bin:/usr/ucb:/etc:/bin:/usr/proc/bin:/opt/lotus/bin
export PATH
Notes_SHARED_DPOOLSIZE=8126464
export Notes_SHARED_DPOOLSIZE
NSD_LOGDIR=/lotus/nsd-logs/notes1
export NSD_LOGDIR
NOTESDATA_DIR=/lotus/notes1/notesdata
export NOTESDATA_DIR
```

Using the variable NSD_LOGDIR is a good practice since it sets the output directory for the NSD script. In this case it would be /lotus/nsd-logs/notes1 for the UNIX user: notes1, change this for each partition, e.g. notes2.

The variable NOTESDATA_DIR defines where the Domino data directory is located on disk.

The Notes_SHARED_DPOOLSIZE is set to optimize the package size of the Domino memory files written to /tmp by the Domino server.

Important: Create a file called .hushlogin in the Notes Data Directory, (for example, by typing: touch .hushlogin). This file disables the OS-Notification a user gets after logging in. (The .profile and .hushlogin files are important for the script provided in “Running your script manually” on page 85).

2.5.3 Network configuration

This section discusses some of the considerations for the network configuration on a Solaris system as they relate to your Domino installation.

Host names and IP addresses

There are a number of different scenarios that can require multiple TCP/IP host names and addresses which may also require additional physical or logical network interfaces (multi-homing and multi-netting). Domino can also be configured with port mapping, which allows an IP address to be shared across partitions.

The following is a best practice for basic Domino servers:

- ▶ Single Domino server - Requires one IP address and host name.
Note: If the Domino host name is not the system's host name, an alias entry must be established tying the two together in the host file's localhost entry.
- ▶ Multiple Domino partitions - We recommend that each Domino partition has its own host name and IP address.
Note: If you do not have enough IP addresses to do this, review the section "Using Port Mapping" on page 117.
- ▶ Clustering - We recommend a private network between the nodes of the cluster, as well as individual host names and IP addresses. As an example, the Domino server Mail01/Acme would have the host names mail01 for the public interface and mail01cl for the cluster interface.

With Solaris, each host name/IP address requires a network interface. This interface can either be a separate *physical* interface or multiple *logical* interfaces on a physical network card.

You may want to use a naming convention that uses a theme for the Domino servers' names as well as the host names. For example, you may want to combine location, function, and instance (such as BostonMail03 or BostonApp02) for the Domino servers' names. For host names, use the Domino server's name if only one IP address is being used. If there is more than one IP address being used you will need to name the host differently per instance. A theme can be useful here as well, such as adding a suffix to the base host name to identify each instance (cl for cluster, pr for private, or wn for WAN). Then link the Domino server's name to this host name in the remote systems name services (i.e. Connection doc, Host file, DNS sub domain or independent DNS domain) to guide the other systems to this IP address as needed.

Tip: Review RFC 1178 "Name Your Computer" for additional guidance in host naming. This material is available at <http://www.ietf.org/rfc/rfc1178.txt>

Important: Do not create host names using the underscore character because it's no longer supported in DNS/BIND, 8.xx and later.

There are a number of different types of network cards that can be configured in a Solaris system. We discuss the setup procedure for PCI network cards that are added to the Solaris system for clustering on an isolated network or for additional network transmission on a Domino partitioned server. (Both of these procedures were discussed previously.)

There are many different Ethernet cards that can be used with the Solaris systems, some with and some without full-duplex support. The following full-duplex Ethernet cards are available:

- ▶ hme or eri- one 10/100 Mb interface
- ▶ qfe 1.0 - four 10/100 Mb interfaces, uses the hme driver and patches
- ▶ qfe 2.0 - four 10/100 Mb interfaces
- ▶ ge - one 1 Gb interface

Restriction: Installing a 1 Gb interface will not mean you get 10 times the throughput of a 100 Mb card. Actual speed is dependent on the speed of the system bus, storage system, and the type of network traffic. Speeds of between 100-200 Mb are common.

Solaris systems come with one Ethernet interface installed. If you need to add additional cards, the following procedures need to be followed for the Domino partition and cluster to work correctly.

Install additional network card

In smaller scale implementations, we recommend using one 100 mbit network card for all the Domino partitions. Use the Solaris logical interface feature (see the online help page - 'man ifconfig') to set up separate hostnames and IP addresses for each partition. Each partition ends up with its own Hostname, IP address, and logical interface. However, in larger scale implementations we recommend using separate network cards for each partitioned server.

The following section describes how to install an additional network card in your system.

First, follow the instructions that came with the network card to physically install the additional card in the system, then use the following steps to configure it.

Note: Usually, all that is required to install a new network card is to put it into the server and issue the **boot -r** command. Solaris will then find the new hardware. If it does find the card successfully you can skip the following steps and continue with "Configuring your network interfaces" on page 33.

1. From the OpenBoot command line (displayed as “ok”) type the following command:

```
ok show-devs
```

The devices installed on the system should be displayed on the screen. Look for the devices that end with “hme.” These are the pci ethernet cards that have been installed. (If you have a QuadFast or Vector Gigabit ethernet card the response will be different. Consult the installation manual for the identification. Other instructions will remain the same.)

2. Using the results, perform the following test:

```
ok apply watch-net <full path name of the hme interface>
```

For example:

```
apply watch-net /pci@1f,4000/SUNW,hme@4,1
```

The following should be the system response to this command:

```
Internal loopback test - succeeded.  
Transceiver check - passed.  
Looking for Ethernet Packets.....  
'.' is a Good Packet. 'X' is a Bad Packet
```

Note: If you do not get the response or your system hangs, reset your system with the power key, interrupt the boot by pressing the STOP key on the Solaris keyboard, and press the A key.

When you are returned to the “ok” prompt, issue the following command:

```
ok reset-all
```

When the system has finished booting, issue the `apply watch-net` command again.

3. Before booting the machine, change the system to recognize each ethernet adapter’s MAC address (internal number assigned to each ethernet card from the manufacturer). By default, the first MAC address is assigned to all additional ethernet devices. Type the following command from the OpenBoot prompt:

```
ok setenv local-mac-address? true
```

This will set the system to recognize each ethernet card MAC address.

Configuring your network interfaces

The Solaris convention for naming network interfaces is as follows:

1. Interface name is a string in the format of the following:
 - *physical-unit*
 - *physical-unit: logical-unit*

For example: hme0 refers to the first hme physical interface, while hme0:1 refers to the logical interface on the first hme physical interface.

- Each network interface needs to know its hostname and IP address. To do this, create a file called “/etc/hostname.<interface-name>.” This file must contain the host name to be associated with this interface name. The first network card “hostname” file is created by the system on install. An example follows.

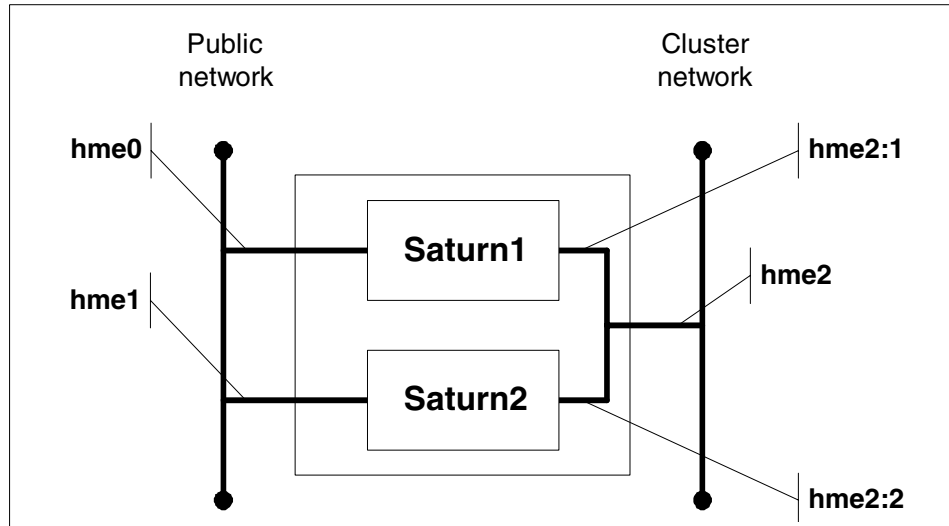


Figure 2-8 Using multiple network interfaces in a cluster

- For our systems we are running two partitions with a private LAN cluster. We have three network cards in our system. The first partition for server “saturn” has a /etc/hostname.hme0 file:

```
root% cat /etc/hostname.hme0
saturn1
root%
```

For our second partition we have a host name of “saturn2” and it has a /etc/hostname.hme1 file:

```
root% cat /etc/hostname.hme1
saturn2
root%
```

For our third network card, which will be used for our private LAN, we have configured two logical interfaces /etc/hostname.hme2:1 and /etc/hostname.hme2:2

```
root% cat /etc/hostname.hme2:1
saturn1c1
```

```
root%
```

```
root% cat /etc/hostname.hme2:2
```

```
saturnc2
```

```
root%
```

You will need to have a valid IP number from your network assigned to the partitioned hostnames. For the IP numbers for the cluster connection on your private LAN, you can select a private IP setting from the list set aside by the IETF (note that these are non-routable on the Internet):

- 10.0.0.0 - 10.255.255.255 (Class A)
- 172.16.0.0 - 172.31.255.255 (Class B)
- 192.168.0.0 - 192.168.255.255 (Class C)

An example that could be used is 192.168.0.XX where XX is the last octet from the IP address for the server.

For example, our server “saturn1” IP address is 9.95.35.68. The last octet is “68” so we will give for our clustering interface the IP address 192.168.0.68.

4. Edit the /etc/hosts file and add the hostnames and IP numbers of all the interface names on the system. Also, because the cluster network is a private network, you must add the IP addresses and host names of all machines on that private network.

Example /etc/hosts file:

```
# cat /etc/hosts
#
# Internet host table
127.0.0.1      localhost
9.95.33.68     saturn1.acme.com saturn1 loghost
9.95.35.53     saturn2.acme.com saturn2
192.168.0.68   saturnc1
192.168.0.53   saturnc2
```

5. Connect the systems that will be accessing each other over the cluster network together. If you have just two clustered systems, the easiest way to connect them is with an ethernet cross-cable. If there are more than two systems, the cluster network cards should be connected to a hub or switch.

Tip: A cross-over cable has pin assignments of:

- 1 - 3
- 2 - 6
- 3 - 1
- 6 - 2

6. To ensure that your Domino server is not routing IP packets, use the following command to create an empty file called `/etc/notrouter`.

```
root% touch /etc/notrouter
```

For information on adding additional network interfaces refer to, “Network configuration” on page 113.

2.5.4 File system layout

The final step in preparing for the installation of your Domino server is to set the file system layout for the Domino binary and data files. During installation of the Solaris Operating System, you sectioned your primary disk for the file systems needed for the Solaris installation. We now will need to do the same thing for the Domino installation.

We have already discussed the considerations for the amount of disk space needed for your Domino implementation, based on a number of parameters. When defining your file system layout, we recommend that you follow the guidelines listed here for performance and reliability. Note that these guidelines apply to locally attached storage; they will need to be adjusted for other types of storage, such as Storage Area Networks (SANs).

- ▶ Use stripes (RAID 0 or 0+1) for optimal performance.
- ▶ Protect your data (RAID 0+1, 1 or 5), even in Domino clusters.
- ▶ Use at least one file system per partition.
- ▶ Put each partition's transaction log file on a separate physical disk.
- ▶ When creating a disk stripe, use disks of the same type.
- ▶ For RAID1 implementations, allocate (at least) one hot spare for each side of the mirror.
- ▶ When creating the Transaction Log File system, use the following parameters for large sequential access:

```
newfs -i 200000 -c 200 -C 15 physical-disk-description
```

Example: `(/dev/dsk/c0t1d0s3)`

Figure 2-9 on page 37 shows the “saturn” system that we implemented in the lab. Notice that the `notesdata (/lotus/notes1, /lotus/notes2)` directories and the transaction log directories `(/lotus/translog/notes1, /lotus/translog/notes2)` *are mounted on separated disks*. The `/opt/lotus` directory is mounted from the `/opt` directory on the primary disk partitioned during the Solaris installation.

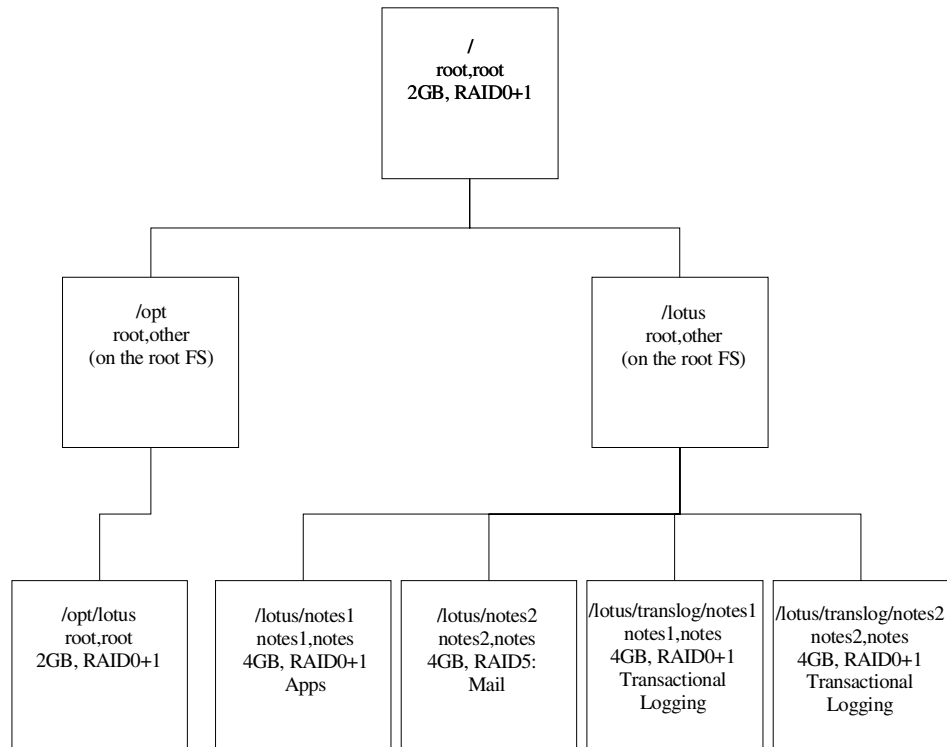


Figure 2-9 The Solaris file system layout we used for the Redbook

With the information provided, you should now be ready to begin the installation of the Domino server. If you are migrating from an NT server to a Solaris server, be sure to read the next section, covering some operating system considerations, before you proceed. If you will not be migrating from another operating system, you can skip directly to the R5 installation instructions in Chapter 3, “Installing Domino R5 on Sun Solaris” on page 43.

2.6 Moving applications from NT to UNIX

Domino databases are *platform independent*, meaning that you can copy files from NT to UNIX and open the database without any kind of change to the file format. However, there are a few considerations to bear in mind due to the differences in the environment.

To ensure that your application will be compatible, consider the following questions before moving an application from NT to Solaris:

1. Is your Domino application “self contained?”

2. Did you use CASE (Computer Aided Software Engineering) tools?
3. Does it use OS platform exploitation?

Is your Domino application “self-contained?”

A self-contained application runs entirely inside the Domino server, without any explicit references to files, without external calls, and without importing or exporting data. An explicit reference to a file, such as `c:\domino\data\NAMES.NSF`, will not work on Solaris and needs to be replaced with `/domino/data/names.nsf`. Solaris does not support the `\` character for specifying paths and uses the `/` character. Solaris is case-sensitive when specifying paths and filenames, while NT is not. Case sensitivity can also be a problem anywhere an external script call, link, or hotspot is used; be sure to check that the correct case is used.

Did you use CASE tools?

While CASE tools may be helpful, many of these tools were created with non-UNIX operating systems in mind and their output code may not be compatible with Solaris. Be sure to check with the manufacturer for compatibility before using these tools.

Does your application use OS platform exploitation?

Anything in the application that might be platform-specific could fail in the Solaris environment. NT-specific services, NT Registry Sync for user registrations, Active-X controls, or compilers that rely on platform-specific libraries to compile the application will cause problems when the application is moved to Solaris.

2.6.1 Moving the application to the Solaris server

Transferring the files from NT to UNIX can be done using many methods. FTP, transfer via CDRW, Iomega Jazz drives, or other media and PCNFS are all good ways of getting the data moved over. For this example we will use FTP, since it is the most common tool used in the field.

Since FTP servers are installed by default on the UNIX side and not on the NT side, it is usually easier to open an FTP session from the NT box and connect to the UNIX box. Here we are using NT 4.0.

1. From the NT box, open an MSDOS command prompt by selecting **Start -> Programs -> MSDOS**.
 - a. Change directory to the server's data directory with the command
`cd \lotus\domino\data`
 - b. Start an FTP session with the command `ftp servername`
2. Change directory on the UNIX box to the data directory with the command
`cd /lotus/notes1/notesdata` (or any other data dir).

Switch to binary transfer mode by issuing the command `bin`.

3. Transfer the databases by issuing the command `put names.nsf`, or, transfer multiple files at once using wildcards with the `mput *.nsf` command.

Important: Never add or remove databases from the OS level while the Domino server is up and running. Domino caches the data directory listing and unpredictable behavior can occur if you modify the data directory while the server is running. This could result in a server crash or hang.

Ensuring permissions are correct

After the transfer is complete, make certain that permissions are correct on the UNIX machine. Log on to the UNIX machine and change to the data directory (`cd /local/notesdata`) and check the permissions on the transferred file `ls -l *.nsf`. An example of the permissions line is:

```
-rw-rw-r-x 1 notes1 notes 1589248 Feb 22 09:34 log.nsf
```

Interpret this record as follows:

The first column shows the permissions. The leftmost letter indicates whether this is a directory or a file. A dash (-) in the left position indicates it is a file; a directory is designated by the letter `d`. The next nine letters indicate the access rights to the file for the owner, group, and world, given in 3 character segments. From left to right the permissions in each segment are read access, write access, and execute access. Therefore an entry of `rwX` means that read, write, and execute access is granted. If any of the letters have a - in their place, then that permission is not allowed. For example, `r-x` means that read and execute access is given but write access is not.

The owner is the user ID that owns the file, which is indicated by the third column in a `ls -l`. The owner's permissions are read from the first three permission characters in column 1 (following the file or directory indicator).

The group is identified in the fourth column. In this case it is the "notes" group. The group's permissions are identified in the next three characters in column 1.

The world is anyone else who has a login access to this system. Their permissions are specified in the last three characters of column 1.

Since the Domino server is the only one that should be changing or directly reading the databases, and databases are not executable programs, the permissions for databases should be:

```
-rw----- 1 notes1 notes 1589248 Feb 22 09:34 log.nsf
```

If the permissions are not correct you can issue the command

chmod 660 filename

where filename is the name of the file on which you wish to change the permissions. This will give read and write access to the database for the Notes user, but will not allow anyone else to view it. Since the Domino server runs under the Notes user account and makes all of the read and write calls on behalf of the clients, most organizations will want to keep the access to the files restricted to the Notes user account.

Checking for case sensitivity

In NT, filenames are not case sensitive, but in UNIX they are. If your scripts call for the file log.nsf and the file is listed as LOG.NSF at the OS, the file will not be found when the script runs. After the FTP completes, check to ensure that the filenames are in lowercase unless your application is specifying otherwise.

```
ls -l
-rw----- 1 notes1 notes 1589248 Feb 22 09:34 LOG.NSF
mv LOG.NSF log.nsf
ls -l
-rw----- 1 notes1 notes 1589248 Feb 22 09:34 log.nsf
```

Important: There are two modes of file transfer in FTP: binary and ASCII. Binary transfers are an exact copy and no reformatting of the file is done by FTP. ASCII transfer assumes the file you are transferring is a text file and, when transferring between platforms, will attempt to reformat the file to the native text format of the destination machine. If you are in ASCII mode when transferring a database, the database will be unreadable by Domino on the destination machine. Some versions of FTP start in ASCII mode. Therefore you should always type **bin** on the FTP command line to ensure that you are in binary mode *before* transferring any databases or templates.

2.7 Summary

In this chapter we have discussed the things that should be considered to properly configure a Solaris server to run Domino R5. You should determine the size and amount of hardware that will be required to properly support your client base.

Consideration should be given to partitioning and clustering as a way to a highly available and scalable server for Domino applications. Proper purchase of disks now can provide less headaches in the future. Allocate sufficient swap space and place swap areas on separate disks when possible. To protect your data, implement some sort of RAID technology, with correct partitioning of your file system.

Once you have installed the Solaris OS, be sure to modify the environment to work correctly with your Domino implementation.

When you have completed all the steps described in this chapter, you should be ready to install your Domino application.

Important: When a file such as the Notes.ini is edited with a Windows editor, you will often find control characters like ^M when the file is transferred to Solaris. The script provided in Appendix F, “Example script to start and shut down a Domino server” on page 395 will automatically transform the Notes.ini to the UNIX standard, but other files have to be checked manually.



Installing Domino R5 on Sun Solaris

In this chapter we describe the installation of Domino R5 on Sun Solaris.

In the first part of the chapter we review the steps necessary to prepare your Sun Solaris environment. Next, we take you step-by-step through the installation of the Domino server code. Then we describe how to set up the Domino server, and finally, we discuss various methods for starting and stopping the Domino server.

In brief, setting up Domino R5 on your Solaris server requires the following steps:

1. Set up the hardware.
2. Install and patch Solaris. (Refer to Domino release notes for required patches.)
3. Set up and test the network.
4. Add and set up the Solaris group and accounts for Domino. (This was covered in Chapter 2.)
5. Install the Domino software from CD-ROM.
6. Set up and configure Domino.

The last two steps are the primary focus of this chapter.

3.1 Prerequisites checklist

Before starting your Domino server installation, you have to:

- ▶ Install and configure the Solaris environment.
- ▶ Prepare the Domino installation.

In this section we provide a checklist for each of these processes. Use them to ensure that you have completed all the steps necessary for a successful Domino installation. Detailed discussion about the preparation process is in the previous chapter.

3.1.1 Install and configure the Solaris environment

1. Check your hardware configuration.
2. Check patch level; if necessary, install the latest patches. You can use `showrev -p` to view the latest patches.

See the Domino release notes (readme.pdf on your Domino CD-ROM) for minimum required patch levels for your release of Domino.

Check Kernel parameters. See readme.pdf on your Domino CD-ROM for current Lotus recommendations.
3. Check Solaris security.
4. Check services. Disable unnecessary services (sendmail, httpd, and so forth).
5. Adapt Solaris configuration files:
 /etc/inetd.conf
 /etc/defaultrouter
 /etc/hosts
 /etc/system
 /etc/profile
6. Prepare TCP/IP.
 - Hostnames (Domino server's common name)
 - TCP/IP address(es)
 - Network mask
 - Domain name server (DNS)
 - Default gateway
 - Static routes (if necessary)
7. Configure network.

3.1.2 Prepare the Domino installation

1. Determine the number of partitions.
2. Prepare the file system. Determine the data directories and mount an appropriate file system for each partition.
3. Determine the destination disk of the Domino binaries. Make sure you have enough disk space.
4. Reserve additional space for transactional logs. As discussed previously, transactional log files must be installed on a separate physical disk. (See Section 2.2.3, “Disk space” on page 14 for details.)
5. Determine user and group names for every partition. For security reasons, we recommend that you not use *notes* for the user name. But notes should be used for the group name.
6. Create a group for the Domino server.
7. Create user accounts for Domino servers. Create additional user accounts when running partitioned servers. Make accounts members of the same group.
8. Configure the UNIX user environment for Domino users. Put Domino data and binary directories into the Solaris account's path environment variable. Use the .profile discussed in the previous chapter. Additionally, edit the .profile or .cshrc, depending on which shell you've selected, and the .hushlogin files.
9. Decide on a Domino naming theme for your organizational units, organizations, and servers. (For example, BosMail01/Sales/Acme in the Acme domain.)
10. Register Domino server IDs for all Domino partitions being installed if these are additional Domino servers in your domain. If you plan to automatically start your Domino servers within a script, do not set a server password when registering the server.

For more information, see Chapter 2, “Sizing and configuration of Solaris” on page 9.

3.2 Install Domino server code

There are two parts to installing Domino on Sun Solaris. The first part is to install the Domino server code, which is done at the server console. The steps to install the code are provided in this section. The second part is to configure the server, which is done via a Web browser interface communicating with the Domino HTTP server. This process is described in the 3.3, “Setting up Domino servers” on page 57.

To run the UNIX program to install the Domino server, perform the following steps.

1. Log in as root.

You log in as root because the files in your Domino program directory (/opt/lotus) should be owned by root.

2. Insert the Domino CD-ROM.

3. Change to the Solaris Directory. Type:

```
cd /cdrom/notesr5/sol
```

Note: The path on the CD-ROM may be different depending on the version of Domino you are installing.

4. Run the install program. Type:

```
./install
```

The screen shown in Figure 3-1 appears.

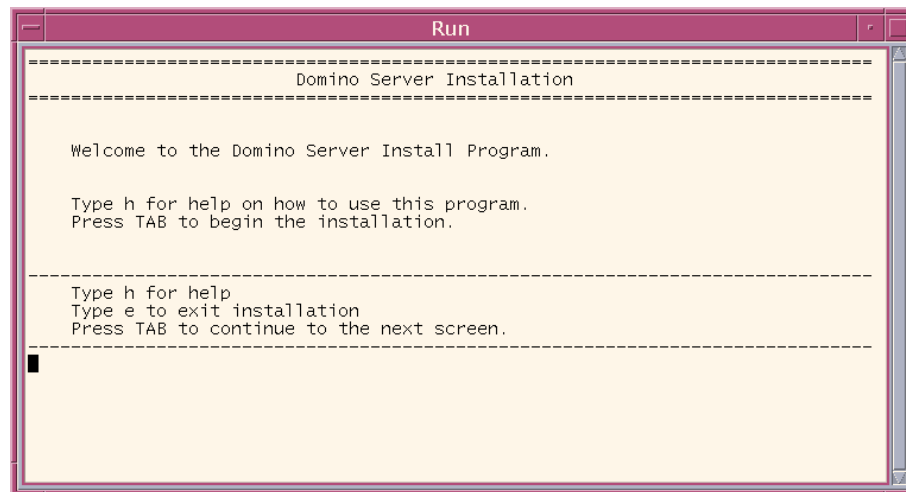


Figure 3-1 Installing Domino: Screen 1

5. Press the Tab key to begin the installation. The installation program displays the Lotus Domino/Lotus Notes Software Agreement. Figure 3-2 shows the first page of the Lotus Domino/Lotus Notes Software Agreement.

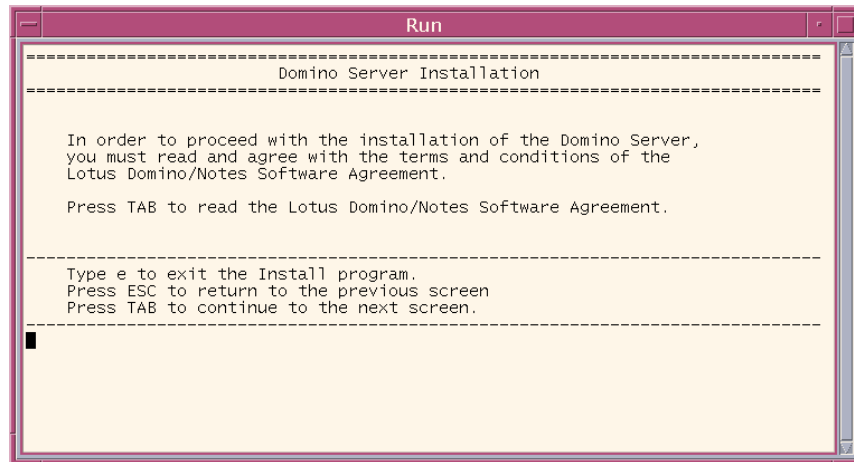


Figure 3-2 Installing Domino: Screen 2

6. Press the Tab key to continue to the next screen, which continues the display of the Lotus Domino/Lotus Notes Software Agreement. Press any key to view each page of the license agreement. Use the Tab key to leave the last page.
7. Figure 3-3 shows the last page of the Lotus Domino/Lotus Notes Software Agreement.

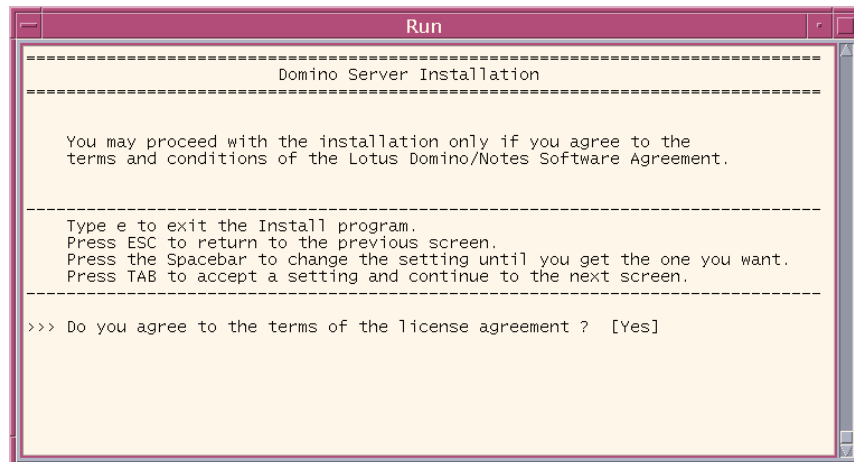


Figure 3-3 Installing Domino: Screen 3

8. Press the Tab key if you agree to the terms of the license agreement. If you disagree, press the Spacebar to select No, and press Enter.
9. If you've accepted the terms of the license agreement, you'll see the screen shown in Figure 3-4.

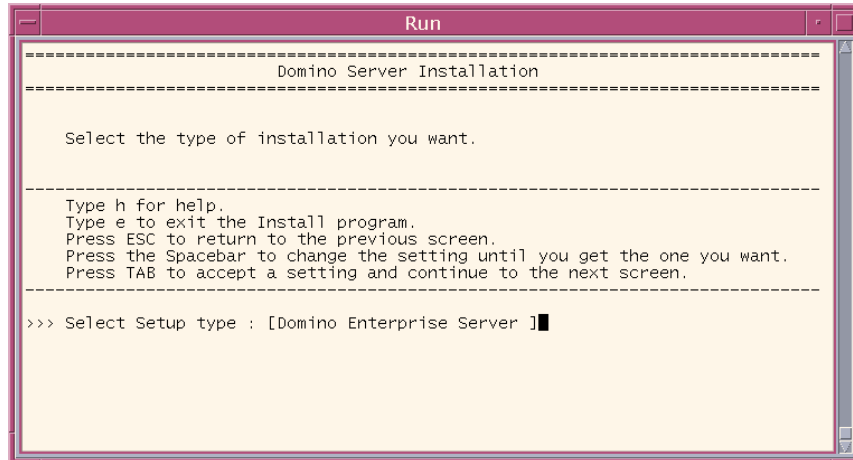


Figure 3-4 Installing Domino: Screen 4

Select the type of setup. Use the Spacebar to scroll through the choices. You can choose between:

- Domino Mail server
- Domino Application server
- Domino Enterprise server (required for advanced services such as partitioning, clustering, and billing).

Press Tab to accept a setting and continue to the next screen.

10. Specify the program directory setting for the destination disk for the Domino binaries. The default is /opt/lotus. Make sure you specify an existing disk that has at least the minimum amount of free disk space necessary.

Press Enter to specify a new location. Press Tab to accept a location.

Note: When you edit the default location, the installation program creates a symbolic link to /opt/lotus. Do not remove this link.

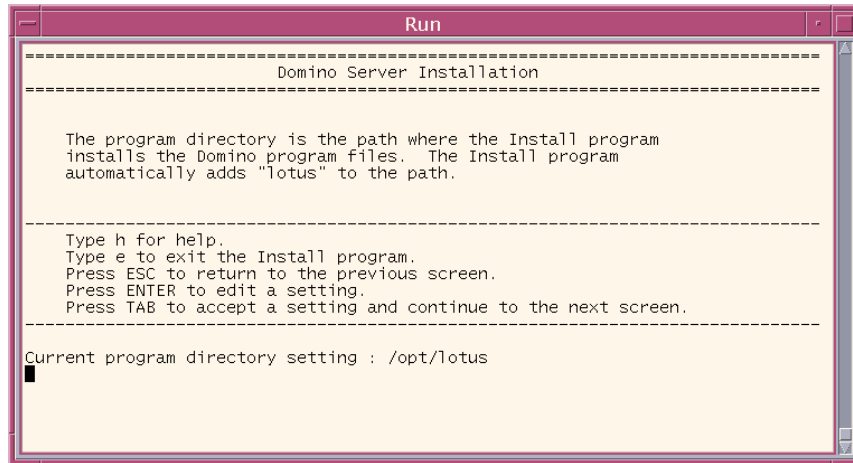


Figure 3-5 Installing Domino: Screen 5

11. After specifying the location for your program directory, you'll see the screen shown in Figure 3-6. Press Tab to continue to the next screen.

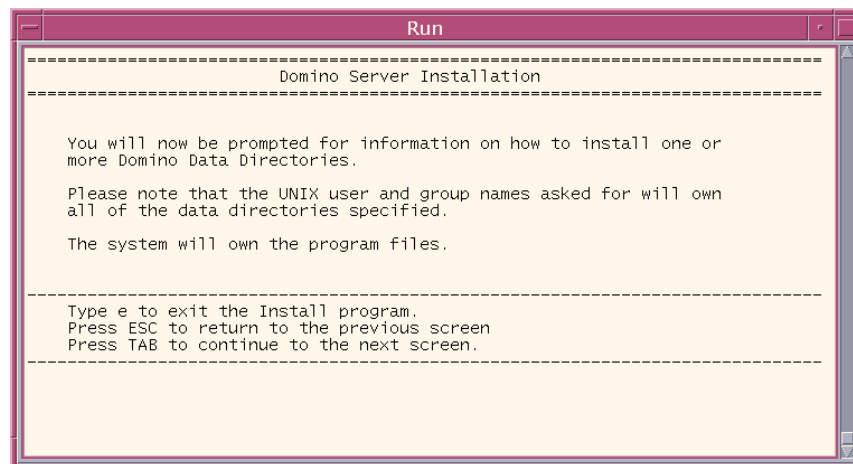


Figure 3-6 Installing Domino: Screen 6

12. Specify whether you want to install multiple partitions:

- If no, press Tab.
- If Yes, press Spacebar and then Tab.

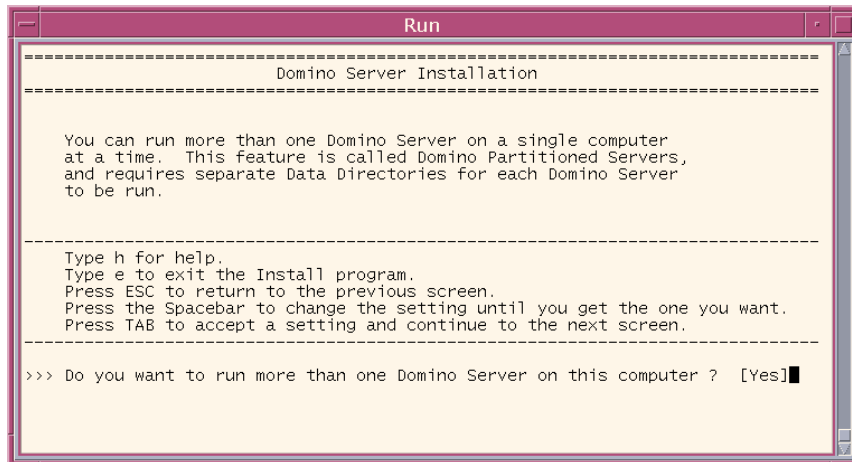


Figure 3-7 Installing Domino: Screen 7

Note: This installation describes how to install Domino partitioned servers. If you decide to install only one server, you can skip steps 13 and 14 and continue with step 15.

13. When you've answered the question "Do you want to run more than one Domino server on this computer" with Yes, the screen shown in Figure 3-8 appears.

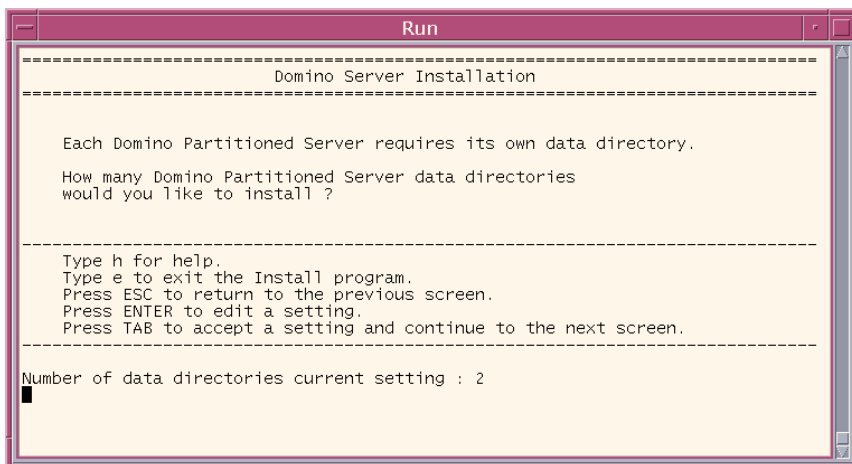


Figure 3-8 Installing Domino : Screen 8

14. The default setting for number of data directories is 2.
 - Press Tab to accept the current setting.
 - Press Enter to edit the setting. The screen for editing the number of data directories appears.
15. Specify the data directory path. You'll be prompted several times to enter the pathname of your data directories, depending on the number of Domino partitions you entered in the previous screen. The default entry when installing only one partition is /local/notesdata. The default entry when installing several partitions is /local/notesdata1.

Notes

We recommend that you separate the Solaris home directory from the Domino data directory. For example:

- /lotus/notes1/notesdata for the first Domino partition's data directory.
- /lotus/notes2/notesdata for the second Domino partition's data directory.

Make sure you specify an existing disk that has at least the minimum amount of free disk space necessary.

Make sure the destination directories for the Domino data directory (notesdata) and Domino program directory are different. Do not install both the program files and the data files in the same directory.

Tip: If you have multiple Domino partitions for your server, it's required that each Domino partition uses a different Solaris Account ID. Each Solaris account ID will have its own Solaris home and Domino data directory.

Example: We have established two partitions on our system for the Domino servers, with the following accounts and directories:

Domino Partition	Solaris Account	Solaris Home Directory	Domino Data Directory
1	notes1	/lotus/notes1	/lotus/notes1/notesdata
2	notes2	/lotus/notes2	/lotus/notes2/notesdata

16. Press Enter to specify a new location.

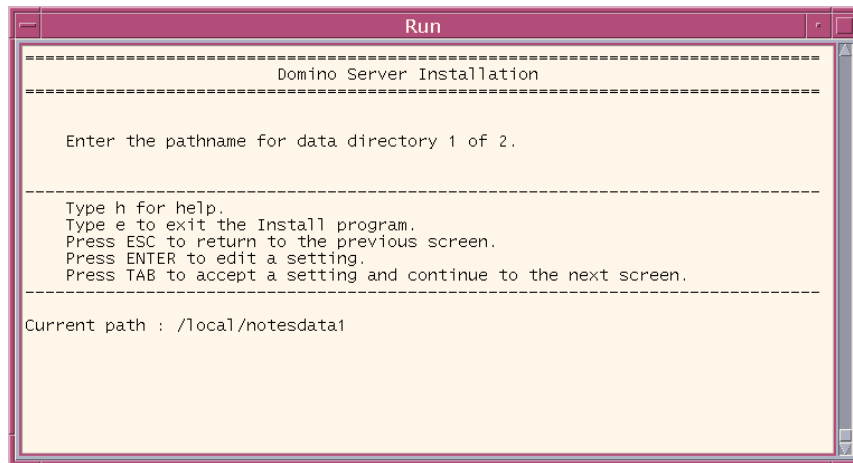


Figure 3-9 Installing Domino: Screen 9

17. Type the new path name and press Enter.

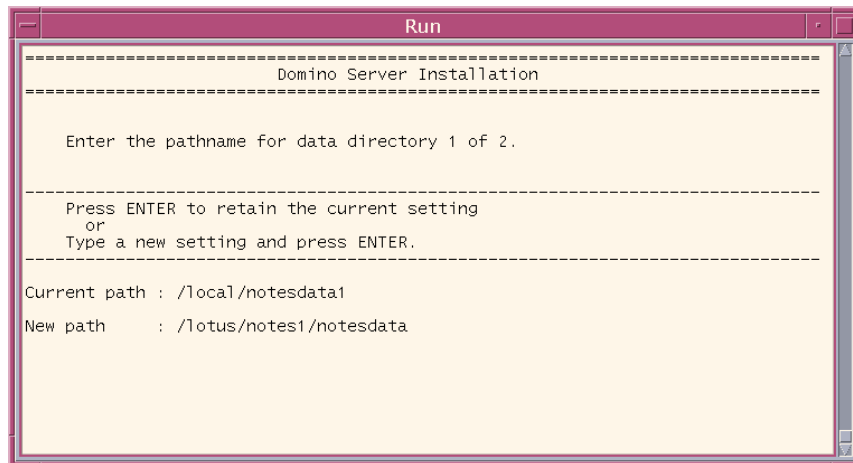


Figure 3-10 Installing Domino: Screen 10

18. The next screen shows the new path that you entered. Press Tab to accept this setting.

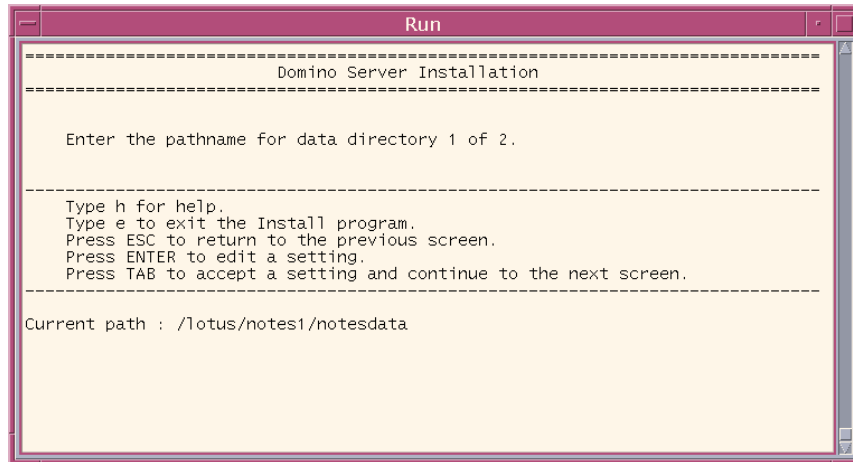


Figure 3-11 Installing Domino: Screen 11

19. Specify the Domino UNIX user name. The user name should be an existing operating system account and requires read/write access to the data directory pathname specified in step 16.

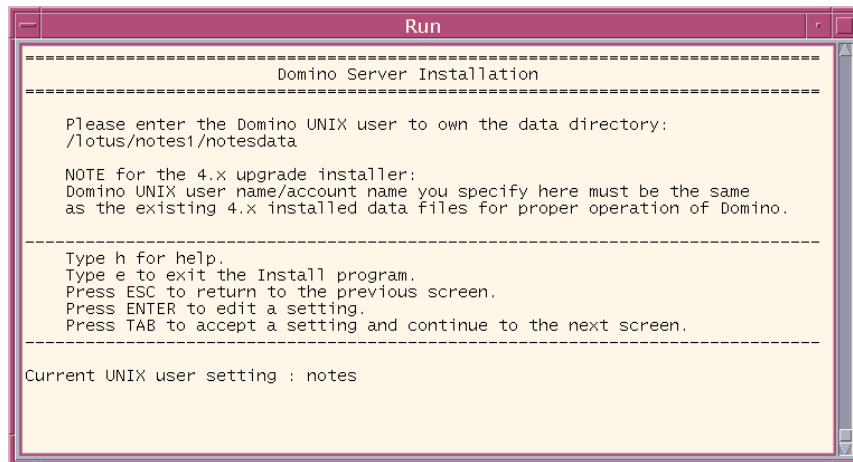


Figure 3-12 Installing Domino: Screen 12

20. Press Enter to specify a new UNIX user name.

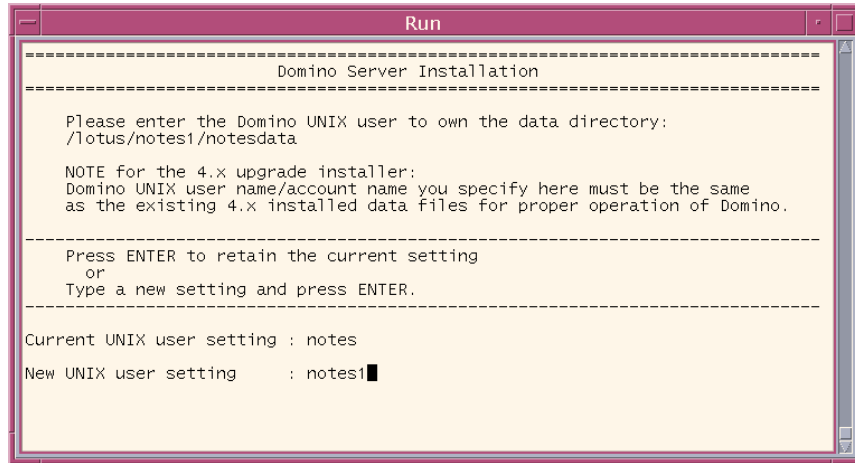


Figure 3-13 Installing Domino: Screen 13

21. Enter the new UNIX user name and then press Enter. The next screen shows the UNIX user name you entered. Press Tab to accept the UNIX user name.

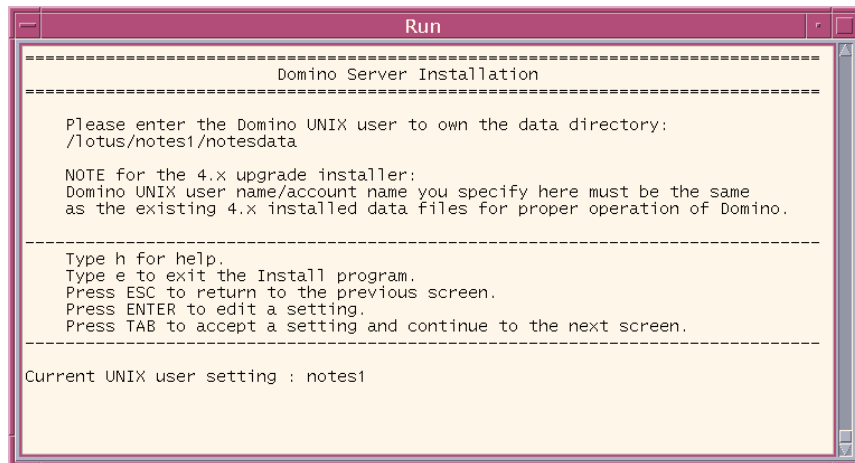


Figure 3-14 Installing Domino: Screen 14

Specify the Domino UNIX group name. The user name specified in step 18 must be a member of this group. Press Enter to edit or Tab to keep the default UNIX group setting.

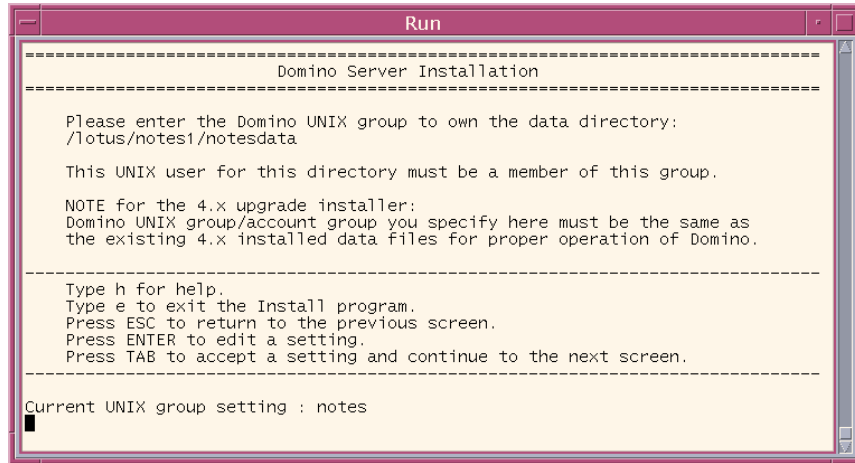


Figure 3-15 Installing Domino: Screen 15

Note: When installing a Domino partitioned server, Steps 15 through 22 are repeated for each partition, changing the data directory and Solaris account each time.

22. At this point, you have finished specifying the installation settings. The screen shown in Figure 3-16 appears. Press Tab to continue to the next screen.

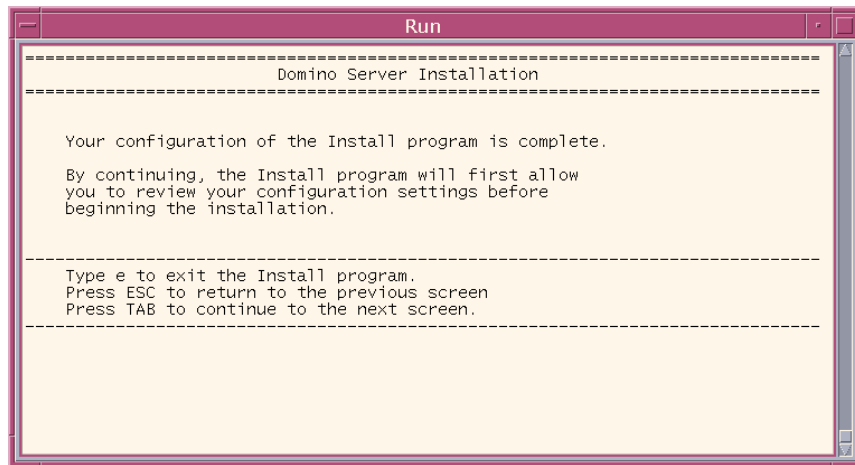


Figure 3-16 Installing Domino: Screen 16

23. Review the installation selections on the screen shown in Figure 3-17. Press Esc to change settings or press Tab to perform the installation.

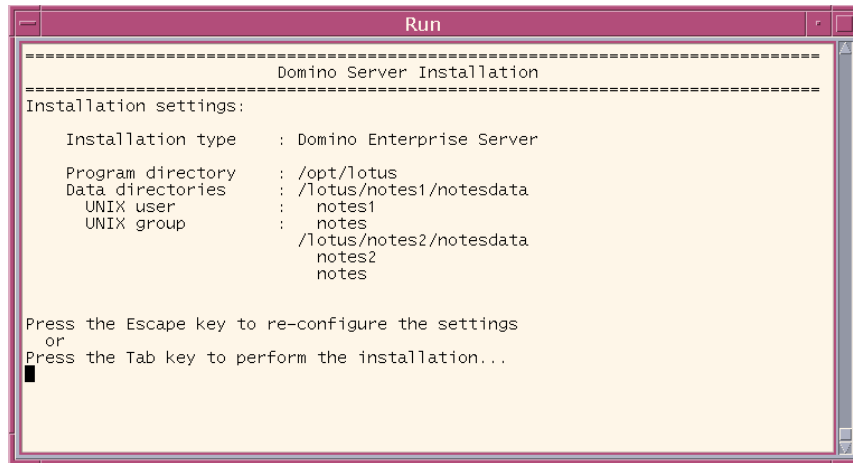


Figure 3-17 Installing Domino: Screen 17

24. Installation is complete.

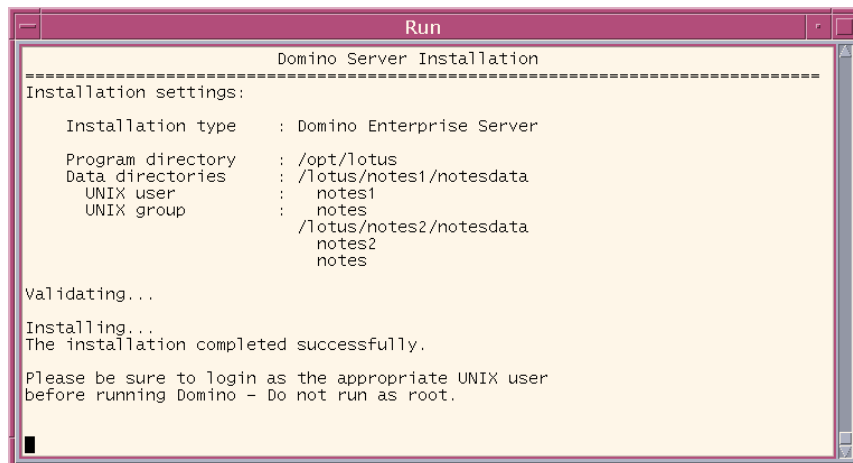


Figure 3-18 Installing Domino: Screen 18

25. Log out of the root account.

Important: Running the Domino server as root is not supported. The files in /opt/lotus directory should be owned by root. This is the reason why you change to the root user ID to install the Domino code. The data directory files should be owned by the default Domino user account created for that partition, that is, /lotus/notes1/notesdata owned by notes1 and /lotus/notes2/notesdata owned by notes2.

3.3 Setting up Domino servers

In this section we describe how to set up your Domino servers. The setup for the first Domino server is somewhat different than that for any additional servers in your configuration. The procedures for both kinds of setup are included here. We also explain how to rerun the setup in case of a failure.

The recommended method to set up your Domino server is via a browser that supports Java and JavaScript, such as Netscape Navigator or Microsoft Internet Explorer. (Do not use the Hotjava browser since it does not support JavaScript.)

The setup can be done either on the server, if it has the appropriate graphics hardware, or from a PC or workstation which has network access to the server. If you don't have a graphical environment, or any workstation which you can use with a browser, you can use the domsetup program. The domsetup program is an application for installing a Domino server using a non-graphical environment. This program is free, but it is not supported by Lotus. For more information, see Appendix D.

Attention: The Domino setup process uses the Java Virtual Machine (JVM) on the machine you are running the browser from. Microsoft Internet Explorer 6.0 and Windows XP do not ship with the JVM installed by default. You can download a JVM from <http://java.sun.com/> or <http://www.ibm.com/developerWorks/>

3.3.1 Setting up the first Domino server

This section shows how to set up the first Domino server in an organization. It is during this stage that your certifier ID and Domino hierarchical naming conventions get established. Section 3.3.2, "Setting up an additional Domino server" on page 64 shows you how to add an additional server in an existing Domino domain. After installing the software code, you can configure the Domino server using a browser client and the Web Setup database (setupweb.nsf).

Perform the following steps to run the setup program.

1. Log in to the system as the UNIX user for the partition you are setting up, as specified in the installation program; for example, notes1.
2. Change to the Domino data directory for the partition you are setting up, as specified in the installation; for example, /lotus/notes1/notesdata.
3. Type the following on the command line and press Enter.
`/opt/lotus/bin/http httpsetup`
4. You will see the screen shown in Figure 3-19, saying that the HTTP Web server has started.

```

$ cd notesdata
$ /opt/lotus/bin/server

Lotus Domino (r) Server, Release 5.0.8 , June 18, 2001
Copyright (c) 1985-2001, Lotus Development Corporation, All Rights Reserved

09/13/2001 12:21:03 PM Database Replicator started
09/13/2001 12:21:08 PM Router: Unable to obtain Internet host and domain names
09/13/2001 12:21:08 PM Mail Router started for domain ACMECORP
09/13/2001 12:21:08 PM Router: Internet SMTP host saturn1 in domain
09/13/2001 12:21:13 PM Index update process started
09/13/2001 12:21:18 PM Stats agent started
09/13/2001 12:21:23 PM Agent Manager started
09/13/2001 12:21:23 PM AMgr: Executive '1' started
09/13/2001 12:21:28 PM Mail01/Acme is the Administration Server of the Domino
Directory.
09/13/2001 12:21:28 PM Administration Process started
09/13/2001 12:21:33 PM Schedule Manager started
09/13/2001 12:21:33 PM SchedMgr: Validating Schedule Database
09/13/2001 12:21:33 PM SchedMgr: Done validating Schedule Database
09/13/2001 12:21:38 PM Calendar Connector started
09/13/2001 12:21:43 PM Event Monitor started
09/13/2001 12:21:48 PM JVM: Java Virtual Machine initialized.
09/13/2001 12:21:49 PM HTTP Web Server started

```

Figure 3-19 Starting HTTP to install a new Domino server

5. Connect to the server via port 8081 with a Web browser which supports Java and JavaScript, such as Netscape Navigator or Microsoft Internet Explorer.

Note: Do not use the Hotjava browser since it does not support JavaScript.

6. Open the browser window and type:

`http://hostname:8081`

where *hostname* is your server's hostname, for example:

`http://saturn:8081`

`http://saturn.acme.com:8081`

Instead of the hostname, you can also use your server's IP address, for example:

`http://9.95.65.68:8081`

7. The server displays the Domino Web Setup database. The first screen of the Domino server configuration is displayed on your Web browser, asking if this is the first server, or are you adding an additional server to an existing domain. In this example, we will configure our first server. (The steps for setting up additional Domino servers are covered later in this chapter.)

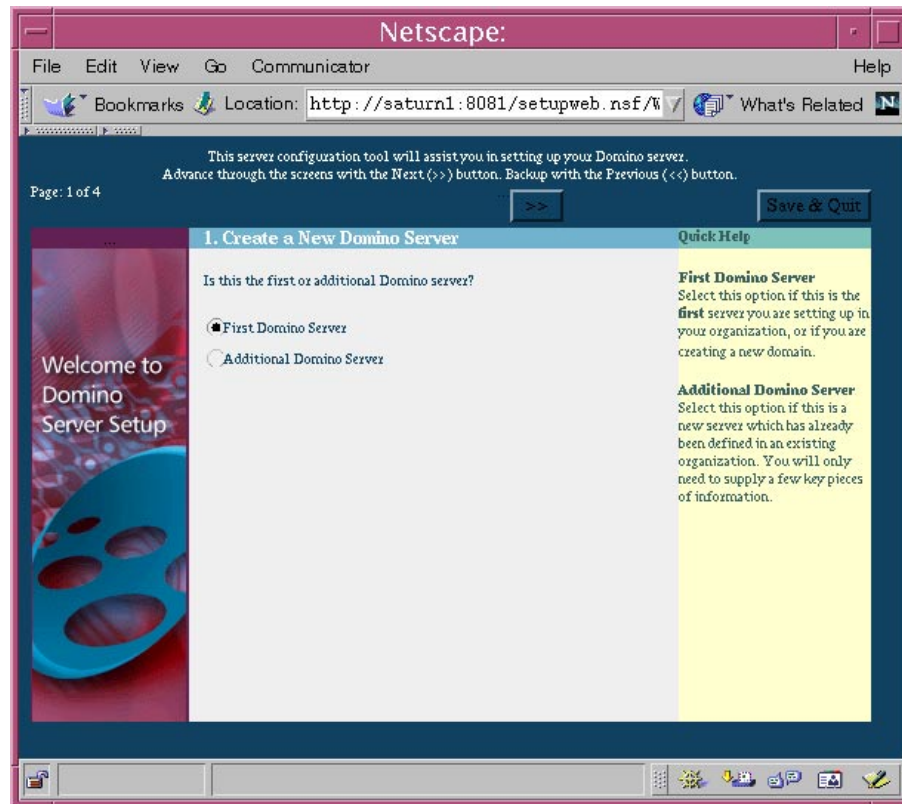


Figure 3-20 Creating the first Domino server

Select “First Domino Server” and click Next (>>). Note that the Next button is near the top of the screen.

8. The setup program questions whether any additional services need to be installed. The default choices are Calendar Connector, Schedule Manager, Event Manager, and Statistics. In our example, we installed HTTP.
You don’t have to install all services during setup. You can activate and configure the appropriate server tasks later.

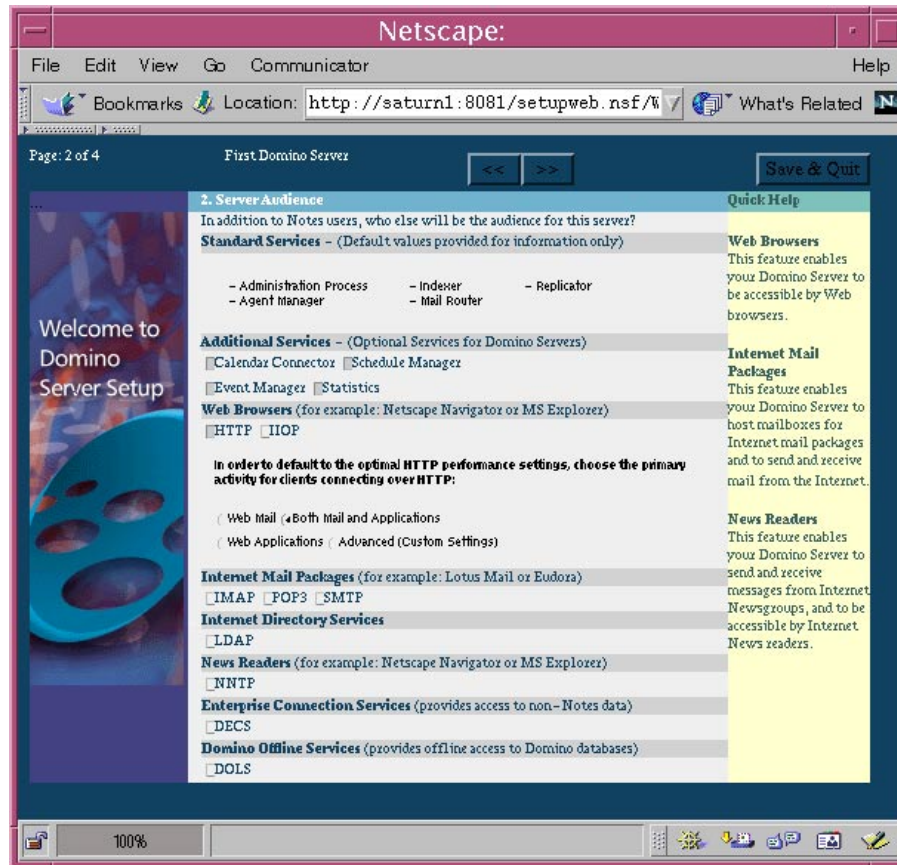


Figure 3-21 Selecting the functions for the first Domino server

Click Next to continue to the next screen.

9. Specify the administrative information for the Domino server. This includes organization name, domain name, certifier name, server name, and the administrator's name.

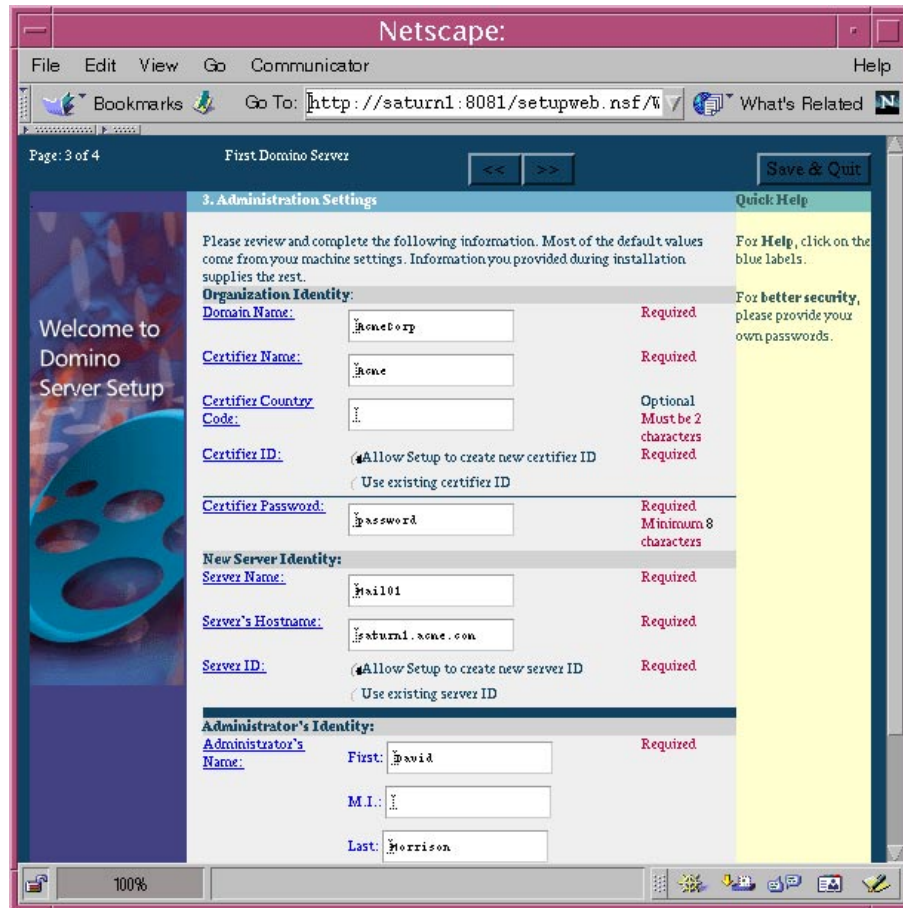


Figure 3-22 Naming the first Domino server and domain

Click Next to continue to the last screen.

Note: Your password is displayed in clear text.

10. Configure the Network and Communication Setting. By default, the “Use all available ports” option is selected. In our example we chose “Use ports selected below”. You should edit the Net Address column to be either the server’s simple IP host name or the fully qualified host name. This name would typically be the Domino servers common name. For example, for Mail01/Acme the simple name would be mail01 and the fully qualified name would be mail01.acme.com.

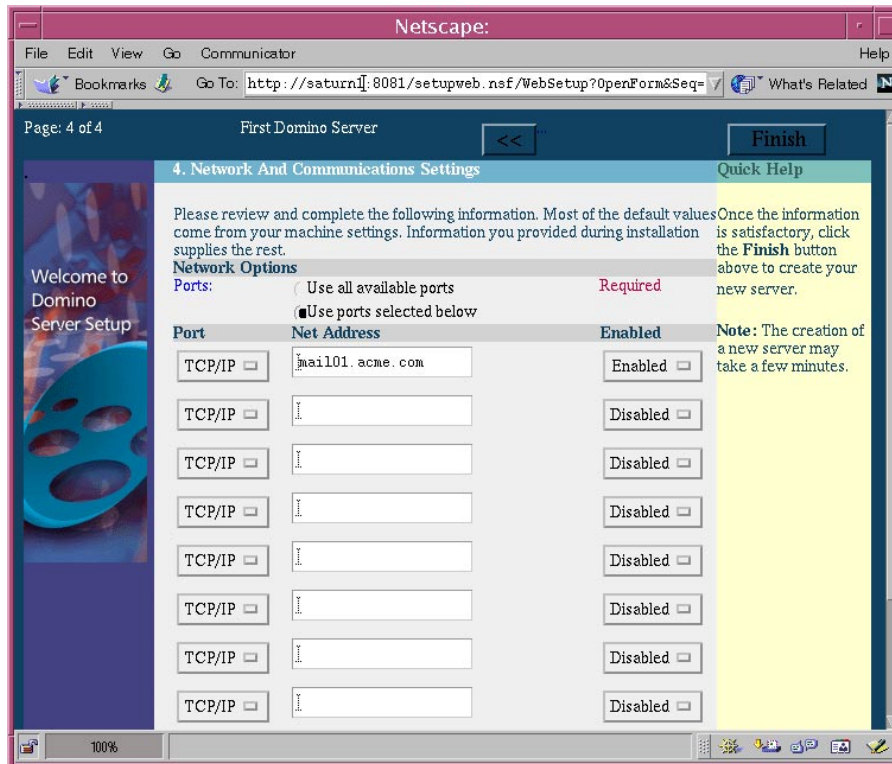


Figure 3-23 Configuring the network ports for the first Domino server

11. Click the Finish button located at the top of the screen.

Note: It is the responsibility of the protocol's name services, for example a Host file, DNS, or connection document to resolve this net address. You will see various status messages displayed on the screen as the server is set up. Click Finish to complete the setup process and you will see various status messages and screens which show the progress of the setup. One of these progress screens is shown in Figure 3-24 on page 63.

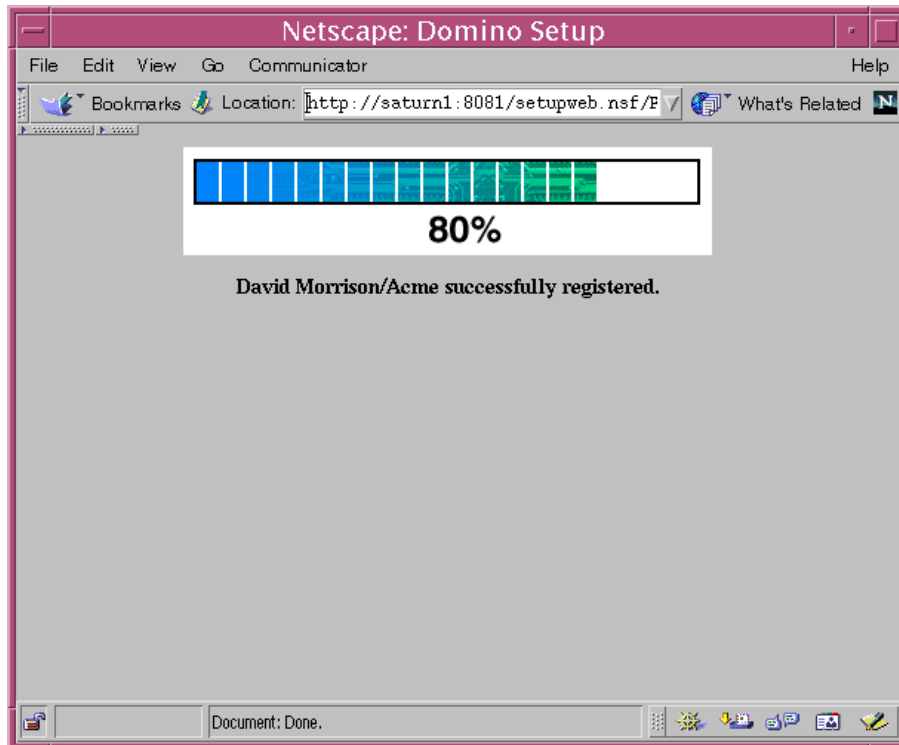


Figure 3-24 Viewing the final steps of the first Domino server setup

Tip: If you do not see this screen, your setup might have failed.

In case of a setup failure, you can find reasons for possible errors logged in the server's Notes.ini file. The Notes.ini file is in your Domino data directory.

12. Finally, you will see the screen shown in Figure 3-25 on page 64.

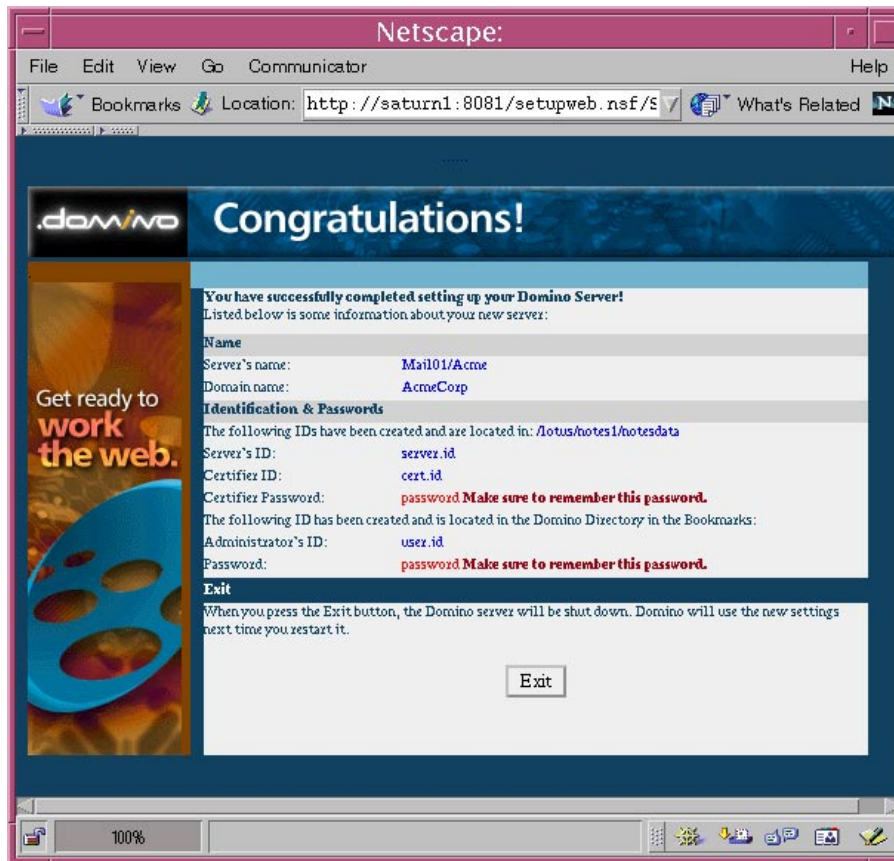


Figure 3-25 Congratulations screen for the first Domino setup

Important: Make a note of the information you see on this screen, especially your passwords.

13. Click the Exit button. The Web server on port 8081 shuts down.

3.3.2 Setting up an additional Domino server

This section describes how you can set up an additional Domino server.

Getting the Administrator ID from the Domino Directory

Before starting the setup process for the second server, you must first register and certify it with the appropriate certifier ID. If you have just configured the first server, the administrator ID will still be stored in the Domino Directory. To get at the ID do the following:

1. Bring the server and http up as normal (see 3.5.1, “Starting the Domino server from Solaris command line” on page 81).
2. Connect with a browser to `http://servername/webadmin.nsf`. You will have to use your admin internet short name (this should be first name and last name, for example “David Morrison”) along with the admin password you specified during the server installation.
3. Click the link titled Directories on the left side, then People at the top of the next screen.
4. Open the Administrator’s user record by double clicking their name.
5. Click the icon user.id as an attachment at the bottom of the record and save the file to your local disk.

Registering a new server

Once you have the administrator ID you need to retrieve the cert.id file from the Domino data directory, `/lotus/notes1/notesdata`, and copy it to the computer where you will be running the Domino Administrator client from. The easiest way to achieve this is to FTP it from the server. Once you have both the administrator.id and cert.id file you can register the second server as follows:

1. Start the Domino Administrator client and if necessary switch to the administrator’s ID file by selecting **File -> Tools -> Switch ID** from the menu.
2. Select the Configuration tab and then click **Registration -> Server** from the menu along the right side of the screen.
3. When prompted, locate the cert.id file and enter the password you set when you configured the first server.
4. In the Register Servers dialog box that is displayed next, click the Registration Server button and enter the name of the first server, for example Mail01/Acme.

- Click the Continue button to display the Register Servers dialog box. Enter a name for the new server such as Mail02 and add the domain information, for example AcmeCorp.



Figure 3-26 Registering a new server

- Leave Password blank. Click the icon labeled “Other” and in the “Store Server ID:” field deselect “In Domino Directory” and select “In File”.



Figure 3-27 Storing the new server ID in a file

- Click Register to register the server in the Domino Directory.

You need your server ID either as a file or attached in a server document in the Domino Directory (names.nsf). If you choose to store the new server ID file in the Domino Directory you will also be required to add a password. Be aware that if you password protect your server ID file, the new Domino server will not be able to automatically start unless you type the password in at the server. To remove the password from the ID file see 3.4.5, “Removing a password from a server ID file” on page 79.

In order to start your Domino servers from a script, we recommend that you register your new server IDs and store them on the file system without a password.

Note: You cannot register additional servers using the Web Admin client; you must use the Domino Administrator client.

Setting up the second server

Make sure that the Domino server from which you want to replicate the Domino Directory is running and that the network connection is up.

The following steps show how to set up an additional Domino server.

1. If you have stored the new server ID file on another machine, FTP it to the Domino data directory, for example `/lotus/notes2/notesdata`.
2. Follow steps 1 through 5 in the previous section (beginning on page 58) to start the setup program.
3. The server displays the Domino Web setup database. On the first Domino server configuration screen, select Additional Domino Server and click Next.

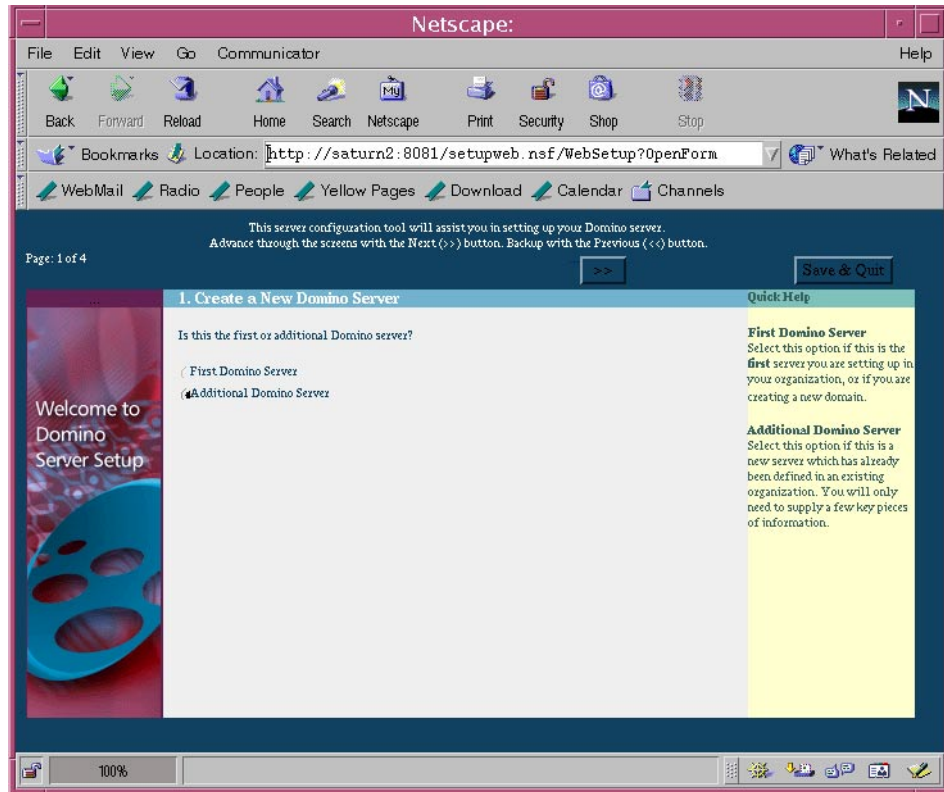


Figure 3-28 Setting up an additional Domino server

4. You are asked whether any additional services need to be installed. The default choices are Calendar Connector, Schedule Manager, Event Manager, and Statistics. In our example, we installed HTTP, IMAP, POP3, and SMTP. You don't have to install all services during setup. You can activate and configure the appropriate server tasks later.

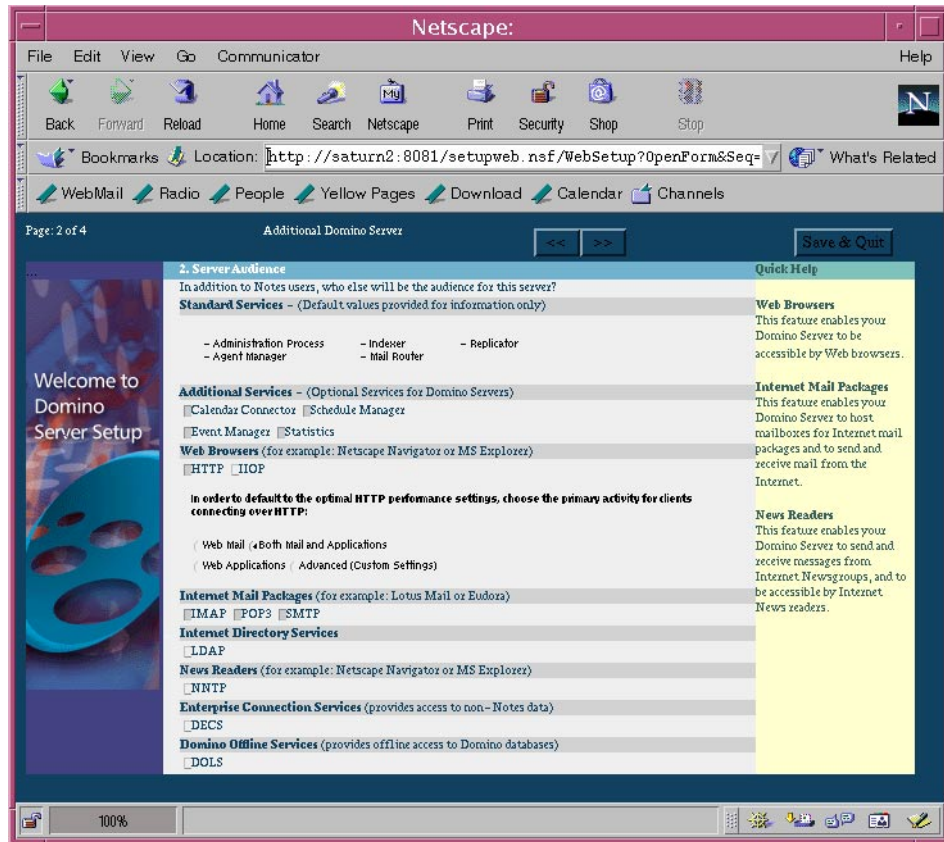


Figure 3-29 Selected services for the additional server

Click Next to continue to the next screen.

5. Specify the administrative information for the Domino server. This includes server name, server's hostname, location of the domain address book and how to connect to the address book server.

Note: If you get your server ID as a file, use FTP to transfer the file in binary format to your local Domino data directory. Don't forget to change the owner and group to your UNIX user and group that is running your Domino server. Select "Server ID supplied in file".

If you stored the server ID file in the Domino Directory when registering the server, select "Get server ID from Address Book". If your newly registered server ID had been attached to the server document a password will be required.

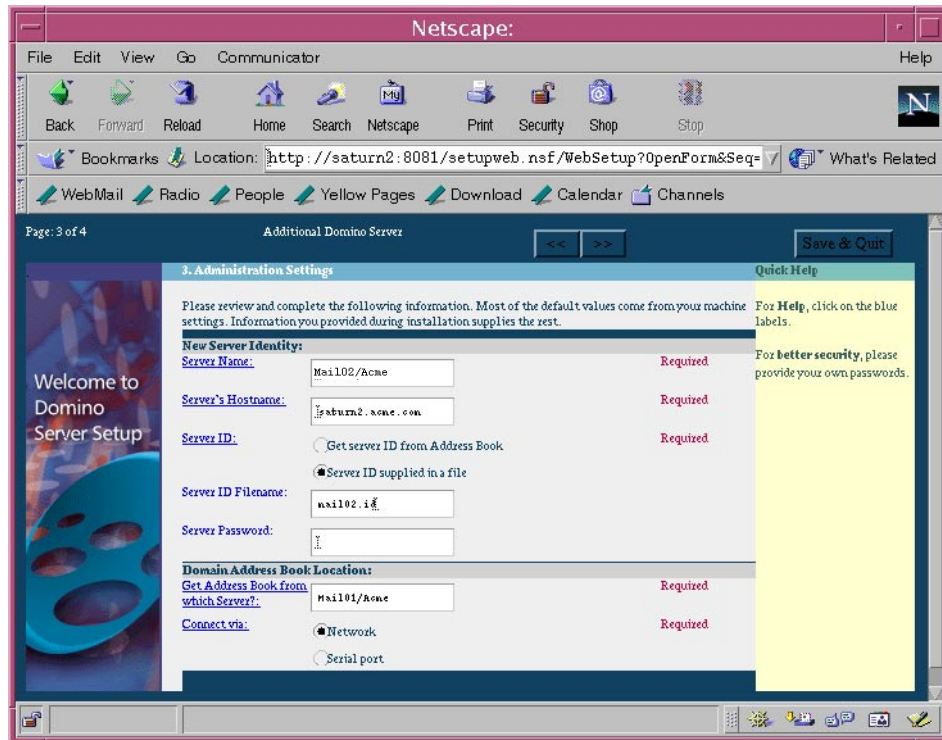


Figure 3-30 Setting the second server name

Click Next to continue.

Note: Enter the complete hierarchical server name in the Server Name field, for example, server2/saturn. Entering only the server's common name will stop the setup process with an error because the server name cannot be found in the Domino Directory.

If the ID has been password protected you will need to specify it, otherwise the setup process will halt. In addition, running the server will require you to enter the password every time the server starts.

To start the server from a script requires that there not be a password on the server ID file. Enter the complete hierarchical server name in the Server Name field, for example, Mail02/Acme. Entering only the server's common name will stop the setup process with an error because the server name cannot be found in the Domino Directory.

6. You should edit the Net Address column to be either the server's simple IP host name or the fully qualified host name. This name would typically be the Domino server's common name. For example, for Mail02/Acme the simple name would be mail02 and the fully qualified name would be mail02.acme.com.

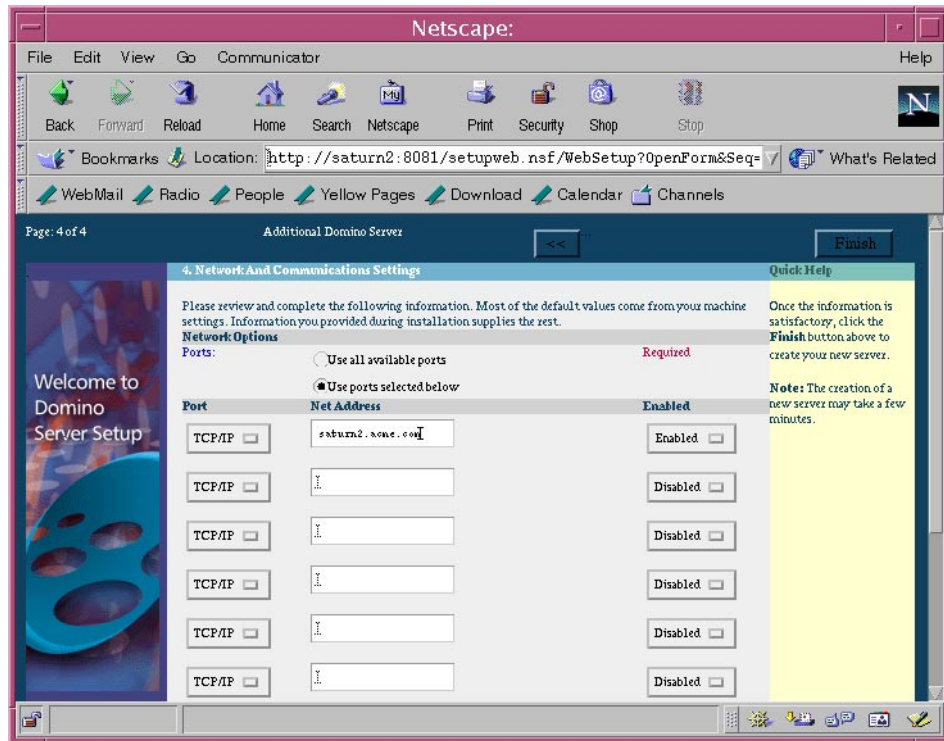


Figure 3-31 Configuring the ports for the second server

7. Click Finish to complete the setup process and you will see various status messages and screens which show the progress of the setup.
Finally, you will see the screen shown in Figure 3-32 on page 72.

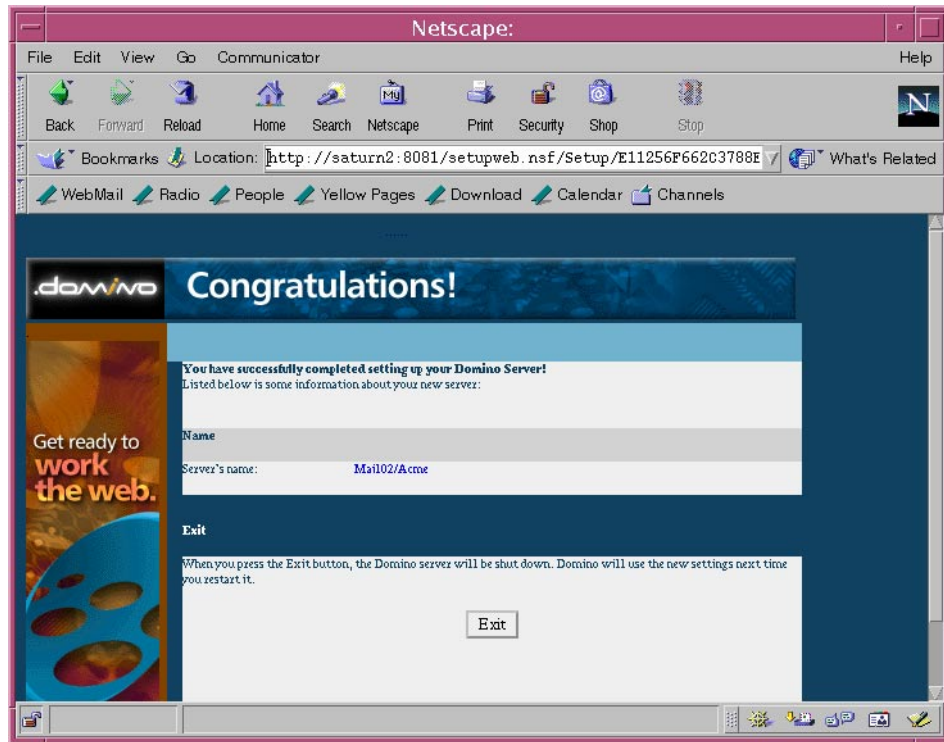


Figure 3-32 Completing the second Domino server setup

8. If you do not see the progress bars displayed before this screen, the setup is likely to have failed, even though it tells you it has succeeded! See the Tip box that follows, and “Rerunning the Domino server setup” on page 73 to restart the installation.
9. Click the Exit button. The Web server on port 8081 shuts down.
10. Since you have now configured more than one Domino server to run on a single server, it is important to configure the Notes NRPC ports before starting the servers. See “Last configuration steps” on page 74, and in particular “NRPC port to IP address binding” on page 75.

Tip: Check Notes.ini for possible error messages. Type:

```
tail notes.ini
```

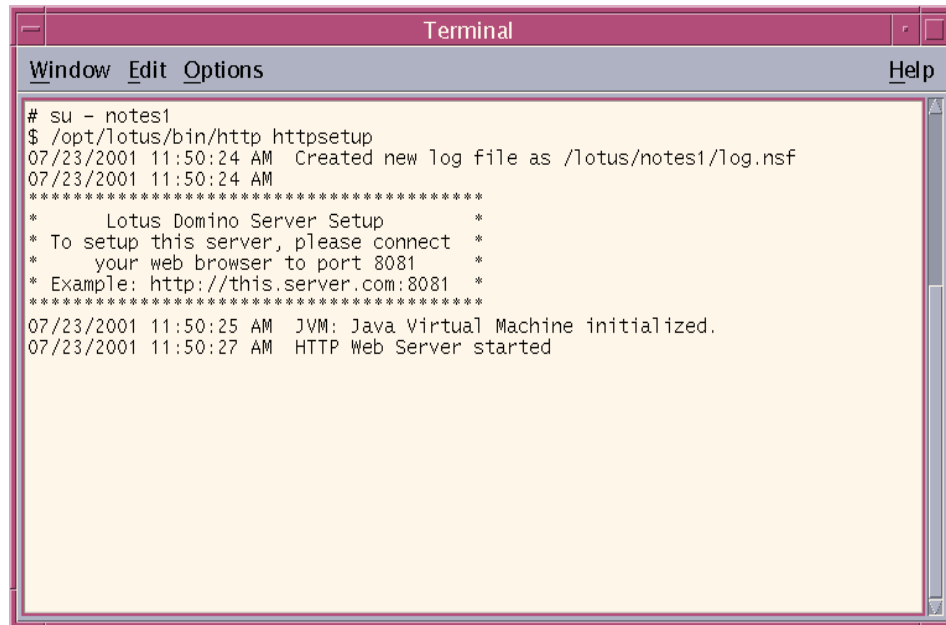
to view the last lines. Error messages always appear at the end of the file. For instance, the following is an example where the second server (Mail02) cannot access the first server (Mail01) to retrieve the Domino Directory:

```
SETUP_PERCENTDONE=20
ExistingServerName=CN=Mail01/0=Acme
KeyFileName=mail02.id
SETUP_ERRORMSG=Server not responding
```

3.3.3 Rerunning the Domino server setup

Use the following steps if you have to rerun the Domino server setup.

1. Log in to the system as the UNIX user for the partition you are installing, as specified in the installation program; for example, notes1.
2. Change to the Domino data directory for the partition you are installing, as specified in the installation; for example, /lotus/notes1/notesdata.
3. Type **/opt/lotus/bin/http httpsetup** and press Enter. You'll see the screen shown in Figure 3-33, saying that the HTTP Web server started.

A terminal window titled "Terminal" with a menu bar containing "Window", "Edit", "Options", and "Help". The terminal output shows the execution of the httpsetup command, log file creation, a welcome message for Lotus Domino Server Setup, instructions to connect a web browser to port 8081, and the successful initialization of the JVM and HTTP Web Server.

```
# su - notes1
$ /opt/lotus/bin/http httpsetup
07/23/2001 11:50:24 AM Created new log file as /lotus/notes1/log.nsf
07/23/2001 11:50:24 AM
*****
* Lotus Domino Server Setup *
* To setup this server, please connect *
* your web browser to port 8081 *
* Example: http://this.server.com:8081 *
*****
07/23/2001 11:50:25 AM JVM: Java Virtual Machine initialized.
07/23/2001 11:50:27 AM HTTP Web Server started
```

Figure 3-33 Startup confirmation screen

4. Connect to the server via port 8081, with a Web browser such as Netscape Navigator or Microsoft Internet Explorer.
5. This should bring up the Congratulations screen in setupweb.nsf with a URL similar to:

```
http://saturn.lotus.com:8081/setupweb.nsf/Setup/F783C49EDCEAAA8D052568A10059EEBE?EditDocument
```
6. Replace the ?EditDocument portion of the URL with ?DeleteDocument and press Enter. The browser should indicate that the document was deleted.
7. Change the URL in the browser again so that it now appears like this:

```
http://saturn.lotus.com:8081/setupweb.nsf
```

The first page of the server setup should now appear.

3.4 Last configuration steps

After you have finished your HTTP setup, you will have to do some final steps before putting your server into production. The required tasks are described in this section.

3.4.1 IP address binding

You have two choices on how to bind the Domino services to an IP interface, either globally across all of the systems' IP addresses or explicitly to a single IP address.

By default, the Domino server, when started, will access all IP interfaces (IP addresses) set up on the system on TCP port 1352.

With one Domino server the binding is global, so all IP interfaces will be bound by this given server on TCP port 1352. When adding additional servers or additional network interfaces, you need to explicitly bind the IP address to the given Notes port, either to different servers or to different ports within a single server.

3.4.2 Configuring partitioned servers

There are two ways to configure the partitioned servers on a computer. You can assign a single IP address to all the partitioned servers and then use port mapping, or you can assign a separate IP address to each partitioned server.

In this section we describe how you can assign separate IP addresses to partitioned servers.

For information on using a single IP address and port mapping, see “Configuring partitioned servers” - “Server Configuration” in the Lotus product documentation (or online help database help\help5_admin.nsf) “*Administering the Domino System Volume 1*”.

The number of partitions is set during software installation. To run partitioned servers on your Solaris system, you need a separate IP address for every server. Each partitioned server's Notes.ini must be edited to include the server's IP address.

The following steps describe how you can assign separate IP addresses to a partitioned server:

1. From the IP addresses you have available, choose one to assign to each partitioned server.
2. For each partitioned server, specify the IP address and the tasks you want.

Note: If you are using a single network interface card (NIC) for the entire computer, Solaris supports multiple IP addresses on an interface card through the use of “logical interfaces.” See Chapter 2 for more information.

3.4.3 NRPC port to IP address binding

There are two means by which the Domino server binds the NRPC port to the IP interfaces (IP addresses) of the system, *globally* or *specifically*.

Global address bind

By default a Domino server binds to all of the available IP interfaces it can find for the single NRPC port it is configured with (TCP port 1352). If the server is exclusive to the system there will be no conflict. However, no other Domino server or other application that uses the same TCP port or set of TCP ports can be started on the system until all Domino servers and/or other applications are explicitly bound to a single IP address per the given server instance. If you have multiple NRPC ports defined per a single server, you will also need to specifically bind each NRPC port to a given IP address with all of the NRPC ports that the server will be configured with.

Note: If the Domino server is configured with other TCP services like SMTP, POP3, IMAP, LDAP, NNTP, or HTTP, you will also need to review their IP address bindings.

Specific address bind

When specific address bind is set, the given Domino server's NRPC port binds to a single IP address exclusively. If there are multiple NRPC ports in use within the given server, the other TCP services (like SMTP, POP3, IMAP, LDAP, or NNTP) will bind exclusively to the first listed port in the PORTS= entry in the Notes.ini. If you are using NRPC cluster replication, it will gravitate to the first listed port in the PORTS= entry in the Notes.ini. In most cases you will want to set the first NRPC port for server-to-server communication. The second NRPC port is then biased for user connections to access, as well as other remote servers. Remember the Notes named network that is in common between the servers will be directly used in any case for mail routing. Server-to-server standard replication connections can be dictated by the selection and pecking order within the given connection document port listing.

Remember that each NRPC port used per a single server (when there is more than one NRPC port) or across the partition servers present on the given system, needs to be specifically bound. In addition, if more than one Domino server is offering the same TCP service (SMTP, POP3, IMAP, LDAP, NNTP, or HTTP), the port will need to be bound exclusively as well.

Configuring the Notes.ini to bind the NRPC port specifically

Binding is done using the NOTES.INI entry <NRPC port>_TcpIpAddress=0,<IP Addr>:<TCP port>, for example:

```
TCPIP_TcpIpAddress=0,9.95.35.68:1352
```

The established NRPC name is used in front of the underscore and the given IP address is that of the IP interface the server will bind to. It is also possible to assign a different TCP port than the default of 1352 by altering the TCP port setting. You should leave the TCP port at the default of 1352 unless:

- ▶ You have a means to redirect the remote systems connection attempt to the new TCP port assignment with either a NAT/PAT firewall system or using the Domino port mapper service.
- ▶ You are explicitly using a connection document which calls out the new TCP port assignment.

Note: You do not need to list the default TCP port 1352 in the entry setting, but it may help remind you what its function is at a later date if you do.

Example 3-1 For a single NRPC port specifically bound

```
Ports=TCPIP
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,130.123.45.1
```


Or, if desired

```
TCPIP_TcpIpAddress=0,130.123.45.1:1352
```

For additional examples, see Appendix E, “” on page 385.

Binding the NRPC cluster replicator service

Unless the first listed port in the Notes.ini PORTS= setting is the ideal port for cluster replication to go over, you will need to add the Notes.ini setting Server_Cluster_Default_Port=<NRPC port> to bias the cluster replication service to the preferred NRPC port. If at all possible, modify the port ordering so the first port listed is the preferred port. Now, the cluster replicator service can roll over to the second NRPC port that the servers share on the Notes named networks they have in common.

If the port is set with the Server_Cluster_Default_Port=<NRPC port> setting and that network has a failure, the cluster replicator will suspend its process until the network is re-established and you will need to use regular replication to resync the clustered databases.

Example 3-2 Cluster replicator bound to a given NRPC port

```
Ports=LBACK, CLUST, TCPIP
LBACK=TCP, 0, 15, 0
LBACK_TcpIpAddress=0,127.0.0.1:2000
CLUST=TCP, 0, 15, 0
CLUST_TcpIpAddress=0,192.101.20.1
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,130.123.45.1
Server_Cluster_Default_Port=CLUST
```

For additional examples see Appendix E, “” on page 385.

Other TCP services IP address binding

As we saw in the previous section, when there is more than one NRPC port present in a given server the other TCP services like SMTP, POP3, IMAP, LDAP, or NNTP will bind exclusively to the first listed port. But this is still a loose binding and may not meet your needs. There are two ways to bind the SMTP, POP3, IMAP, LDAP, or NNTP services with a hard binding. One way is to bind the service to the given NRPC port, which is bound to a given IP address; the other way is to bind the service to its own IP address directly. If the choice between these methods is based on making it easier to maintain over time, the preferred

method is binding to the NRPC port. The HTTP service is handled differently. It is configured in the HTTP section of the server document in the Domino Directory. Table 3-1 show the bindings for each service controlled in the Notes.ini. Note that you can only bind a single IP address per service offering for a given server.

Table 3-1 Other TCP services Notes.ini parameter entries

TCP Service	Binding to NRPC Port	Binding to IP Address
POP3	POP3NotesPort= <NRPC Port> name>	POP3Address=<IP Address or IP Host name>
IMAP	IMAPNotesPort= <NRPC Port> name>	IMAPAddress=<IP Address or IP Host name>
SMTP	SMTPNotesPort= <NRPC Port> name>	SMTPAddress=<IP Address or IP Host name>
LDAP	LDAPNotesPort= <NRPC Port> name>	LDAPAddress=<IP Address or IP Host name>
NNTP	NNTPNotesPort= <NRPC Port> name>	NNTPAddress=<IP Address or IP Host name>

Example 3-3 is for a partitioned server, where the server's single NRPC port is specifically bound to an IP address. This server's SMTP service is then bound to the *same* IP address.

Example 3-3 Specifically binding SMTP service to NRPC port

```
Ports=TCPIP
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,130.123.45.1
SMTPNotesPort=TCPIP
```

The next example is for a single NRPC port specifically bound and SMTP service bound to a *different* explicit IP address.

Example 3-4 Specifically binding SMTP service to explicit IP address

```
Ports=TCPIP
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,130.123.45.1
SMTPAddress=209.98.76.10
```

The next example is for a single NRPC port specifically bound to an IP host name. This host name must be resolved via either a local host file or DNS lookup to the server it is accessing.

Example 3-5 Specifically binding SMTP service to a host name

```
Ports=TCPIP  
TCPIP=TCP, 0, 15, 0  
TCPIP_TcpIpAddress=0,130.123.45.1  
SMTPAddress=smtp1.acme.com
```

Additional information can be found in “Configuring your Notes.ini for partitioned servers” on page 116 and additional examples can be found in Appendix E, “Example TCP port Notes.ini settings” on page 385.

3.4.4 Setting up security for your Domino server

Before putting your server into production, you should set up security for your Domino server. For more information, see Section 7.6, “Protecting a Domino server” on page 188.

3.4.5 Removing a password from a server ID file

It is not possible to automatically restart your Domino server if your server ID file is password-protected. If it does contain a password, you will be prompted for a password each time you start your Domino server.

Note: The following method will work as long as a password policy has not been set on the ID file.

The following steps describe how to remove a password from a server ID file:

1. Make sure you are logged in with the UNIX user you created for running the Domino server.
2. Shut down your Domino server.
3. Change to the Domino data directory.
4. Use FTP to transfer the server.id file to your local machine. Use binary mode for file transfer.
5. Open the Domino Administrator client.
6. Click the **Configuration** tab.
7. Select **Certification -> ID Properties**.

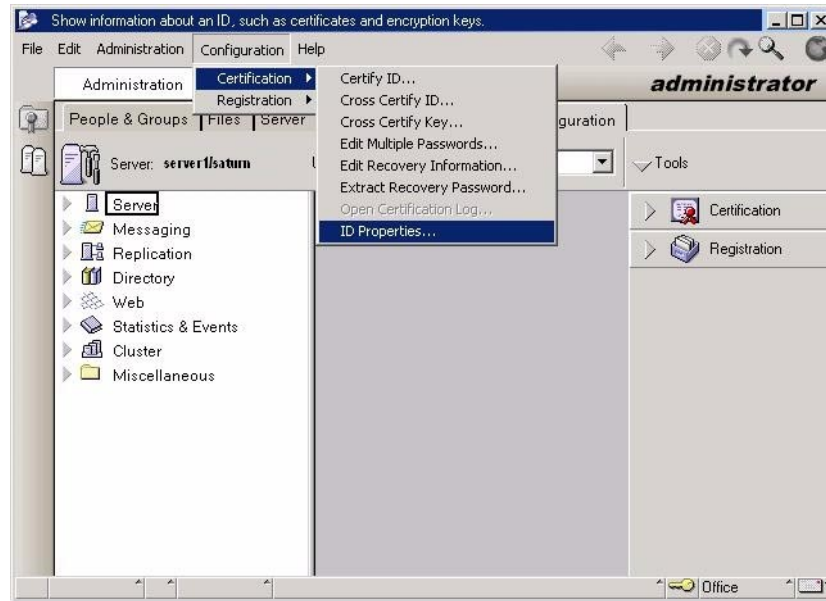


Figure 3-34 Removing a password from a server ID file

8. Open the server.id file from your local drive.
9. Enter the password for the ID file.
10. Click the Clear Password button on the Basics tab to clear the password. The status bar will give you the information "Password cleared."

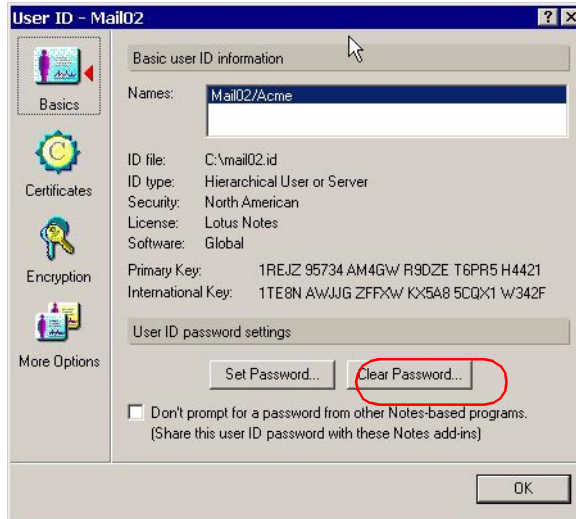


Figure 3-35 Clearing the password from a Domino server ID

11. Click OK to close the ID Properties dialog box.
12. Use FTP to transfer the server.id file back to the server. Use binary mode for file transfer.
13. Make sure that the server.id file has the correct owner and group.
14. Start the Domino server. You will not be prompted for a password.

3.5 Starting the Domino server

This section describes how you can start your Domino server. There are several ways to start a Domino server:

- From the Solaris command line in the foreground and background
- With a startup script

3.5.1 Starting the Domino server from Solaris command line

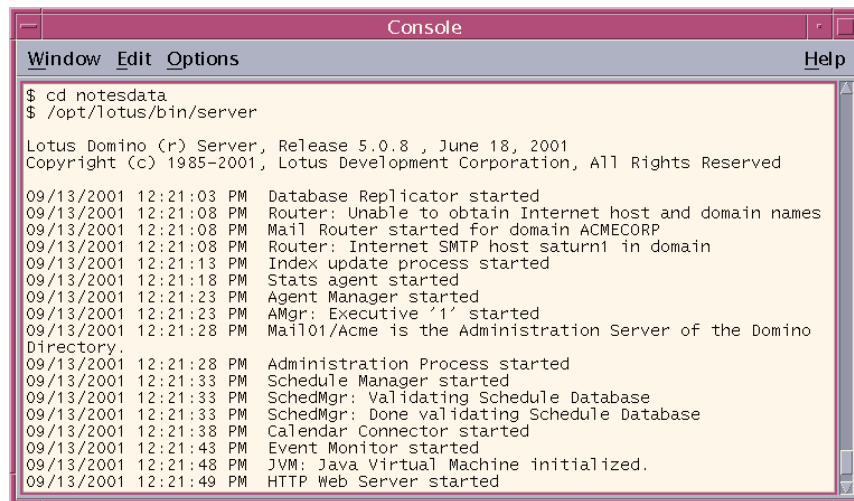
The procedure described here will start the server, and you will be able to view the server console on a screen or window.

Important: If you kill this screen, the server will crash and will not respond to any requests. To recover from this situation, use the `nsd` command to kill the server and all the child processes and remove the corrupted shared memory and semaphores that they created.

To start the server, use the following steps:

1. Make sure you are logged on with the UNIX user you created for running the Domino server, for example, `notes1`.
2. Open a console window and change to your Domino data directory, for example, with the command `cd notesdata`
3. Type:
`/opt/lotus/bin/server`
4. Press Enter.

Figure 3-35 shows some of the messages you will see during Domino server startup.



The screenshot shows a terminal window titled "Console" with a menu bar containing "Window", "Edit", "Options", and "Help". The terminal displays the following text:

```
$ cd notesdata
$ /opt/lotus/bin/server

Lotus Domino (r) Server, Release 5.0.8 , June 18, 2001
Copyright (c) 1985-2001, Lotus Development Corporation, All Rights Reserved

09/13/2001 12:21:03 PM Database Replicator started
09/13/2001 12:21:08 PM Router: Unable to obtain Internet host and domain names
09/13/2001 12:21:08 PM Mail Router started for domain ACMECORP
09/13/2001 12:21:08 PM Router: Internet SMTP host saturn1 in domain
09/13/2001 12:21:13 PM Index update process started
09/13/2001 12:21:18 PM Stats agent started
09/13/2001 12:21:23 PM Agent Manager started
09/13/2001 12:21:23 PM AMgr: Executive '1' started
09/13/2001 12:21:28 PM Mail01/Acme is the Administration Server of the Domino
Directory.
09/13/2001 12:21:28 PM Administration Process started
09/13/2001 12:21:33 PM Schedule Manager started
09/13/2001 12:21:33 PM SchedMgr: Validating Schedule Database
09/13/2001 12:21:33 PM SchedMgr: Done validating Schedule Database
09/13/2001 12:21:38 PM Calendar Connector started
09/13/2001 12:21:43 PM Event Monitor started
09/13/2001 12:21:48 PM JVM: Java Virtual Machine initialized.
09/13/2001 12:21:49 PM HTTP Web Server started
```

Figure 3-36 Starting up the Domino server from the console

3.5.2 Starting the Domino server in the background

From an operational standpoint, running the Domino server in the background is often desirable.

The following steps show a standard way to put the Domino server into the background:

1. Make sure you are logged on with the UNIX user you created for running the Domino server.
2. Change to your Domino data directory, for example, /notesdata
3. Type:

```
/opt/lotus/bin/server &
```
4. Press Enter.

Sending a command to the console

To send a command to the console, you can use the remote console feature in a Notes, Domino Administrator, or Web Admin client. You can also use the Character Console (cconsole.exe) program as described in the following steps.

1. Log on as the UNIX user you created for running the server.
2. Change to your Domino data directory and enter the following on the command line:

```
cd notesdata  
/opt/lotus/bin/cconsole
```
3. You will be prompted for the path to the administrator's ID file, for example, /export/home/notes1/data/jsmith.id.
4. Next you will be prompted for the password.
5. The Domino console prompt ">" will appear, indicating that console commands may now be entered.
6. To end a console session, type "done" at the console prompt.

Attention: Entering **quit** or **q** will shut down the Domino server, although you will be asked if this is what you really want to do before the server shuts down.

3.5.3 Starting the Domino server using a startup script

A complete example script for starting and stopping Domino servers can be found in Appendix F, "Example script to start and shut down a Domino server" on page 395 and can also be downloaded from the ITSO website. (See Appendix H, "Additional material" on page 419 for details.) The following sections refer to this script.

The recommended automated method to start a service on Solaris is to use *run control* (rc) startup scripts. On Solaris the rc scripts are organized in a number of subdirectories. The /etc/init.d directory contains the run control scripts. These scripts are written to accept a **start** argument to start a service and a **stop** argument to stop a service. In /etc there are also run control directories named **rcx.d** where the “x” is a run level. For example, **rc3.d** represents Run Level 3, while **rcS.d** is for Single User. To automatically run a script for a particular run level, links to the script are created in these rcx.d directories. Link names that begin with “S” cause the script to be called with the **start** argument.

Link names that begin with “K” are called with the **stop** argument. After the S or K is a 2 digit number indicating the order the scripts will be executed in. There is no harm in applying the same sequence number to multiple scripts. In this case the order of execution is deterministic, but unspecified. When entering a run level, the K scripts are executed first, followed by the S scripts.

The general procedure to use rc scripts to automate a service is:

1. Create a script which accepts the **start** and **stop** arguments that has the commands required to start and stop the service.
2. Place the script in /etc/init.d and make it “executable” by setting the “x” permission.
3. Create a link that starts with S in the rc directory for the run level you wish to start the service. Often this is done in /etc/rc3.d for services like Domino.
4. Create links that start with K in the run level directories where you want to stop your service. For services like Domino, this is usually /etc/rc0.d and /etc/rc1.d

For more information about run levels and run control scripts see the man pages for **init** and **init.d**.

A Domino script example

To link this script with multiple filenames perform the following steps:

1. Create your script.
2. Log in as root and copy the script to the /etc/init.d directory:
cp name-of-script /etc/init.d/lotus

Tip: When you transfer your script from DOS to UNIX in binary mode, the script file has to be converted from DOS format to ISO format. Enter **dos2unix lotus lotus** to convert the format. You don't have to convert the format if you use ASCII transfer mode.

3. Add execute permission:
chmod +x /etc/init.d/lotus
4. Create hard-links with the Solaris ln command:
ln /etc/init.d/lotus /etc/rc3.d/S99lotus
ln /etc/init.d/lotus /etc/rc1.d/K00lotus
ln /etc/init.d/lotus /etc/rc0.d/K77lotus

Running your script manually

Before running the script, make sure you have created the .hushlogin file in each Domino server's home directory. (See "UNIX environment for Domino user ID" on page 30 for details.)

You can run your script manually by performing the following steps:

1. Log in as root.
2. Change to /etc/init.d by typing:
cd /etc/init.d
3. To start the Domino servers type:
./lotus start
4. To stop the Domino servers type:
./lotus stop
5. To start a single Domino server type:
./lotus start notesx e.g. (notes1)
6. To stop a single Domino server type:
./lotus stop notesx e.g. (notes1)

3.6 Shutting down the Domino server

This section describes how you can stop your Domino server. There are several ways to stop a Domino server:

- ▶ From the foreground server console
- ▶ From the Domino Administrator client
- ▶ From the Solaris command line
- ▶ With a stop script

If you have a problem shutting down, use the NSD command with the -kill switch. See Section 12.3.1, "Running NSD" on page 312 for more information.

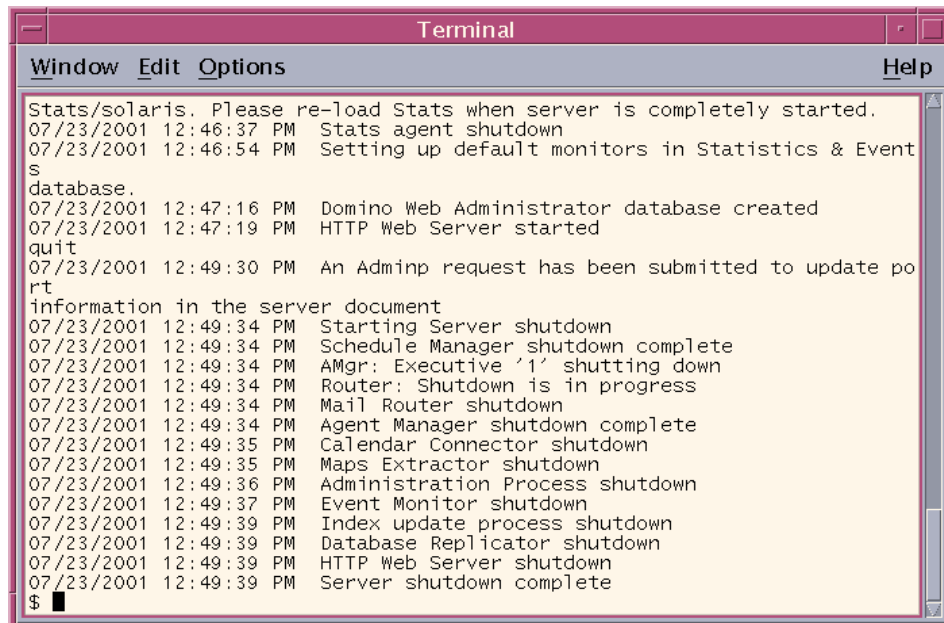
Important: You should not use the Unix “kill -9” command to stop the Domino server since this will not clean up the used resources and allow Domino to be restarted.

3.6.1 Shutting down from a foreground server console

When your Domino server is running in the foreground, you can perform the following steps to shut down the server:

1. Type:
`exit`
or
`quit`
2. Press Enter.
It may take a few seconds or more for the server to shut down.

Figure 3-37 shows the console during shutdown.



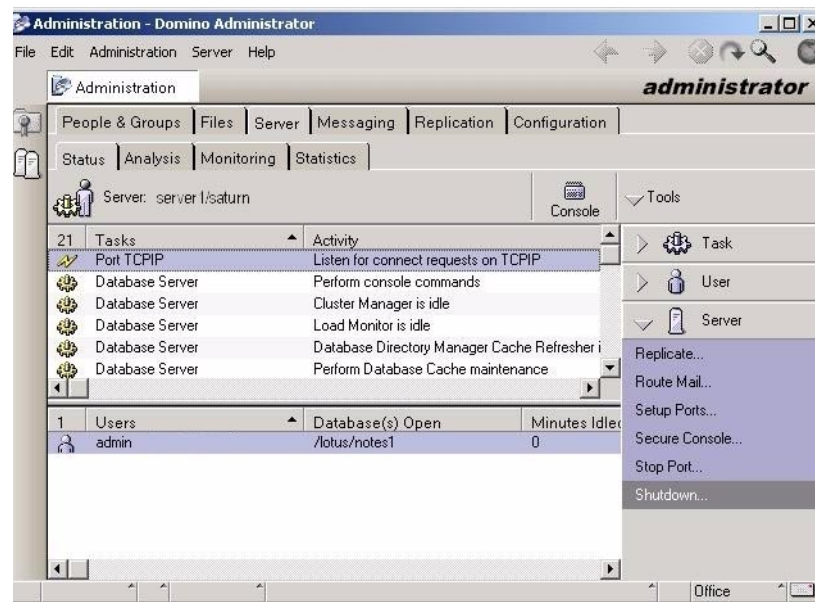
```
Stats/solaris. Please re-load Stats when server is completely started.
07/23/2001 12:46:37 PM Stats agent shutdown
07/23/2001 12:46:54 PM Setting up default monitors in Statistics & Event
S
database.
07/23/2001 12:47:16 PM Domino Web Administrator database created
07/23/2001 12:47:19 PM HTTP Web Server started
quit
07/23/2001 12:49:30 PM An Adminp request has been submitted to update po
rt
information in the server document
07/23/2001 12:49:34 PM Starting Server shutdown
07/23/2001 12:49:34 PM Schedule Manager shutdown complete
07/23/2001 12:49:34 PM AMgr: Executive '1' shutting down
07/23/2001 12:49:34 PM Router: Shutdown is in progress
07/23/2001 12:49:34 PM Mail Router shutdown
07/23/2001 12:49:34 PM Agent Manager shutdown complete
07/23/2001 12:49:35 PM Calendar Connector shutdown
07/23/2001 12:49:35 PM Maps Extractor shutdown
07/23/2001 12:49:36 PM Administration Process shutdown
07/23/2001 12:49:37 PM Event Monitor shutdown
07/23/2001 12:49:39 PM Index update process shutdown
07/23/2001 12:49:39 PM Database Replicator shutdown
07/23/2001 12:49:39 PM HTTP Web Server shutdown
07/23/2001 12:49:39 PM Server shutdown complete
$
```

Figure 3-37 Console during shutdown

3.6.2 Shutting down from the Domino Administrator

Another way to shut down the server is to use the R5 Domino Administrator client. Use the following steps to start the Domino Administrator client and shut down the server. For more information on using the Domino Administrator client, refer to the (XXX-add x-ref to chapter Administration)

1. Open the Domino Administrator client.
2. Click the Server - Status tab.
3. Choose the server you want to shut down and select **Tools -> Server -> Shutdown**, or from the menu bar, **Server -> Server -> Shutdown**.



4. You will be prompted to confirm the shutdown. If you want to shut down the server, click **Yes**. This will shut down the server.
5. The Domino Administrator will indicate that the server no longer responds.

Tip: You can shut down and restart your Domino server by using the command **restart server** from the console. However, this does not work if the server is in a hung state.

3.6.3 Shutting down from the Solaris command line

There are two ways to shut down the Domino server from the command line.

Method 1

1. Make sure you are logged in with the UNIX user you created for running the Domino server.
2. Make the Domino data directory your current directory and type:

```
cd notesdata  
server -q
```


to shut down the server.

Note: The directory containing the installation binaries, /opt/lotus/bin, must be in your PATH statement.

Method 2

1. Make sure you are logged in with the UNIX user you created for running the Domino server.
2. Execute the cconsole command by typing

```
cconsole
```
3. You will be prompted for the path and filename to the administrator's ID file, followed by the password. (This ID file needs to have administrator rights to the server you are trying to shutdown).
4. Type

```
quit
```

3.6.4 Shutting down the server from a script

You can shut down your server manually using the script from Appendix F, "Example script to start and shut down a Domino server" on page 395.

1. Log in as root.
2. Change to /etc/init.d by typing:

```
cd /etc/init.d
```
3. To stop the all the Domino servers type:

```
./lotus stop
```
4. To stop a single Domino server type:

```
./lotus stop notesx e.g. (notes1)
```

3.7 Summary

In this chapter, we gave a short checklist on what has to be prepared prior to installation of the Domino server. We showed step-by-step instructions for installing the Domino server code as well as for setting up the Domino server using a browser and the Web Setup database.

Finally we showed different methods for starting and stopping your Domino server.



Tuning Domino Server on Solaris

One of the most important tasks in your Domino implementation is the proper tuning of your Domino servers. There are many factors that affect the efficient operation of your Domino server environment in the Solaris OS. Before you can accurately tune your server, your Domino implementation should be complete and you should observe your system in operation with a sizable load applied.

The Sun/Lotus team maintains current tuning information for Solaris and Domino at the following website: <http://www.lotus.com/dominosolaris>

In this chapter we discuss several of the areas to consider when you want to optimize the performance of your Domino application on Solaris. Specifically, the tuning-related topics covered here are:

- ▶ Solaris OS system configurations
- ▶ Network configurations
- ▶ RAID
- ▶ Notes.ini

4.1 Solaris OS considerations

There are four main aspects of the Solaris system that can affect your Domino server performance:

- ▶ Disk I/O
- ▶ Memory
- ▶ CPU
- ▶ Network

This section describes the approach we took in the lab when tuning our Solaris environment with respect to these areas, as well as referencing some suggestions from the book, *Sun Performance and Tuning, Java and the Internet*, by Adrian Cockcroft and Richard Pettit.

Before you begin to tune your system for optimal performance, you must first monitor to see exactly how your system is performing.

There are a number of Solaris utilities that can monitor performance-related factors, including:

vmstat	Reports virtual memory statistics regarding process, virtual memory, disk, trap, and CPU activity
mpstat	Reports per-processor statistics in tabular form
iostat	Interactively reports terminal, disk, and tape I/O activity, as well as CPU utilization
netstat	Shows network status
ps (with options)	Prints information about active processes
sar	System Activity Reporter

Many of these utilities have parameters that allow you to log the output to disk for viewing at a later date.

Knowledge about the Solaris operating system is necessary to be able to work with and understand these utilities. The Sun Web site at <http://docs.sun.com/>, as well as numerous references for the Solaris OS, can provide this information.

You can also gather Solaris performance statistics with the Domino R5.08 console command, **show stats**.

The following three tables display the performance measurements considered important, along with notes and recommendations regarding the techniques. Performance measurements should be taken during peak activity times. In addition to normal peak hours, be sure to include measurements from off-hour maintenance times.

Table 4-1 Memory

	Solaris command for monitoring	Domino command for monitoring	Threshold
Percent swap used	swap -s		30 % of all swap devices
Memory allocated		This is the total allocated Domino statistic in statrep or "show stats" command.	Amount of physical memory
Memory shared		View in statrep.nsf or "show stats" command. This is the Domino shared pool.	¼ memory allocated to server
Memory scan-rate	vmstat		If a scan rate (sr) consistently greater than 50 for extended periods
r (runable threads)	vmstat		r= steadily or increasing to >2 per CPU

Table notes:

- ▶ These thresholds are maximum values. Measurements in excess of these values indicate a memory shortage.
- ▶ To tune the memory system in case of memory shortage, add more physical memory. For a Domino Server, memory should be sized so as to keep memory scan rates low.

Table 4-2 CPU Usage

Statistic	Solaris command for monitoring	Threshold
% sy	vmstat and mpstat	%sy > 20%
% us	vmstat and mpstat	%us > 60%

Table 4-3 IO Disk activity

Statistic	Solaris command for monitoring	Threshold
Percent of time disk is active	iostat	%b consistently greater than 50% on any single disk volume or file system can create a bottleneck. Disk rates (IOPs) vary and need to be factored in to any evaluation.
Percent IO wait	vmstat and iostat	>20% and increasing. If there is an IO bottleneck this value will increase.

4.1.1 Considerations for clustered servers

- To optimize the memory used to cache the databases in the cluster perform the following on the primary server for each cluster:
 - Open cldbdir.nsf and select any database in the view that should not be replicated via the cluster, such as:
 - *.ntf
 - log.nsf
 - events4.nsf
 - statrep.nsf

(This might include all system databases except admin4.nsf, names.nsf, cldbdir.nsf, clubusy.nsf)
 - Choose **Tools -> Disable Cluster Replication** from the menu. For each selected database the status will be updated to Disabled.
- To set the optimal Mail Router Failover set the following in the Notes.ini file:
- To set the cluster failover for recovery or for load balancing set the following value in the Notes.ini file:

Cluster_MailFailover=2

Server_TransInfo_Normalize=300

This is a tuning setting. If the failover process takes too long, then the value might be too high. If it occurs too quickly, then the value is too low.

Note: See the article in *Iris Today*, "Optimizing Server Performance: Domino clusters (Part 2)" (available at <http://www.notes.net/today.nsf>) for a detailed explanation. Consider the Availability Threshold of 95 for recovery.

4.1.2 Solaris kernel tuning

Solaris kernel parameters are set in the `/etc/system` text file. Software applications may require you to set parameters here to override the Solaris default settings in order for the application to function. You may also decide to set parameters here to tune the system to optimize performance.

Prior to Solaris 8 and Domino R5.0.8, many Solaris kernel parameters were recommended; however, with Solaris 8 and Domino R5.0.8 far fewer adjustments are needed. Consult the *Common Tuning Tips* document available at <http://www.lotus.com/dominosolaris> for the latest guidance.

4.1.3 Solaris file system tuning

Solaris supports a number of different file system types, for example `smfs`, Veritas file system, and so forth. The following parameters are for the default Solaris file system known as `ufs`. Depending on your disk and controller technology, the settings for the `-C` parameter will vary. The settings identified here override the default settings and will give you better performance, but may not be optimal for all disk technologies.

Transaction logging file system parameters

When creating the file system that will contain the transaction logs, use the following command line:

```
newfs -i 200000 -c 200 -C 15 /dev/rdisk/...<your disk name>
```

Domino database file system parameters

When creating the file system that will contain the Domino databases (`.nsf`), use the following command line:

```
newfs -i 200000 -c 200 -C 7 -m 1 /dev/rdisk/...<your disk name>
```

4.2 Network configuration

Important: Good client-server communications involve the client, the network, and the server. If there is a problem, it could be with any one of these components. Experience suggests that simply tuning the server settings is seldom effective without first studying the entire network for performance related issues.

In the first chapter we discussed some of the things that should be done on the network if you are using multiple network cards. In this section we discuss a way to monitor your network for performance, and describe additional tuning that may be required based on your server tasks as well as your network topology or topologies.

The Solaris OS has a utility that monitors the traffic inbound to and outbound from your server. The command to use for this is **netstat**.

By using the different options provided with **netstat**, you can get a good understanding of how your network is handling the Domino traffic and if there is any network bottleneck to contend with.

The following are some basic network changes that can improve your application's performance:

- ▶ Move from a shared segment network to a switched network design.
- ▶ Move from 10 Mbps networks to 100 Mbps configurations.
- ▶ Segregate your traffic types to different networks; that is, local user LAN, remote user LAN, server only LAN.
- ▶ Use multiple LAN segments (one for each partition) to isolate network traffic at the high-end user loads.

One of the netstat options, **netstat -k interface#** can be used to give you a dump of all the kernel statistics. The output from this command will resemble the following:

```
saturn1% netstat -k hme0
hme0:
ipackets 6820075 ierrors 0 opackets 337 oerrors 0 collisions 0
defer 0 framing 0 crc 0 sqe 0 code_violations 0 len_errors 0
ifspeed 10 buff 0 oflo 0 uflo 0 missed 0 tx_late_collisions 0
retry_error 0 first_collisions 0 nocarrier 0 inits 7 ncanput 0
allocbfail 0 runt 0 jabber 0 babble 0 tmd_error 0 tx_late_error 0
rx_late_error 0 slv_parity_error 0 tx_parity_error 0 rx_parity_error 0
slv_error_ack 0 tx_error_ack 0 rx_error_ack 0 tx_tag_error 0
rx_tag_error 0 eop_error 0 no_tmds 0 no_tbufs 0 no_rbufs 0
rx_late_collisions 0 rbytes 1112237011 obytes 40534 multircv 2 multixmt 0
brdcstrcv 6819755 brdcstxmt 3 norcvbuf 0 noxmtbuf 0 phy_failures 0
```

If excessive collisions or packet errors are encountered there may be some areas that need to be tuned on your network. Consult the Sun documentation for more information on TCP and Internet server tuning.

Network tuning on a Solaris system is performed by means of the **ndd** command. Changes are used by new connections immediately. To enable the parameters system-wide you can type them from the command line. To maintain the settings upon reboot you should enter them in the `/etc/rc2.d/S69network-tuning` script.

4.2.1 TCP/IP maximum transmission unit (MTU) sizing

This section presents a review of the different network topologies MTU sizes and how their interaction can impact the reliability of the application data transport between the Notes or Domino servers. You may also encounter problems with TCP/IP MTU sizing depending on the version of the system's TCP/IP stack (end nodes and routers) and the types of devices and topologies that make up your network. This can happen when the end node systems (Domino servers or Notes clients) are located on Token-Ring or FDDI networks directly, located in a mix with Ethernet network segments, or with WAN or SLIP/PPP dial-up connections that you are trying to access across. Work with your network administrator to discuss any possible conflicts, and use the largest value your network can support.

Testing the pathway

Using the ping TCP/IP tool, you can verify what the limitations of the network are. Note that not all ping variations offer the same functions. The ping utility must be able to create variable test packets and set the Don't Fragment flag, which prevents the packet from being fragmented by either the router or the direct end node system. The Windows 95/98/NT version of ping offers these functions. You may also need to test both directions to discover the failure, and you might also need to monitor the router hop since only one pathway direction at a given time may have the problem. Use TRACERT to learn the router hops, and ping hop by hop and in both directions if needed.

Use Table 4-4 on page 98 to base your measurements for the value of the test packet, where the test packet returns acknowledgments for each successful packet, plus one more byte to the test packet, should give you an error indicating the packet needed to be fragmented. This break point is the maximum size the pathway supports. In some cases, the larger test packets return one or two errors. This is not an issue with packet sizing but should be investigated with your network administrators as it is an indication of a general network health problem.

Depending on your TCP/IP stack and your network devices, you may need to set the MTU size manually. Most TCP/IP stacks use Maximum Segment Size (MSS) discovery to learn the local segment (or routed segment) TCP data size, which is then translated into the IP packet size. By default, 576 is used when the TCP/IP MSS can not be discovered. Newer TCP/IP stacks use the MTU path discovery

method to learn the limits of the entire pathway. In some cases, these mechanisms fail to offer the needed constraint or prevent the effective use of the topologies' abilities. Some TCP/IP stacks may need to be manually set or tuned for the local segment topology since their default setting is set to the minimum value (576), which is not recommended for most LAN networks (Ethernet, token-ring, or FDDI). In some cases you may need to add in an additional interface so you can offer local systems the full size possible and remote systems a constrained size that meets the network pathway limits. Make sure the stacks on the network routers are also set to the correct size for the LAN and WAN topologies in use, and any routers with WAN links are set to smaller values. If you need to lock down the MTU manually, use the following guidelines on the needed sizes.

Setting the MTU value

In Solaris MTU is controlled by using the **ifconfig** command, with the syntax:

```
ifconfig <physical interface> mtu n
```

where n is the value you wish to use.

As an example:

```
Ifconfig hme0 mtu 1500
```

Note: If you add additional NICs into your server you will also need to deal with the protocol name resolve services so the different groups of users or servers access the correct interface, as well as create additional NRPC ports and bind each to the interfaces and IP addresses they have so that connections are not misdirected.

Table 4-4 MTU values for different frame types

Topology/ Frame type	Frame size	IP packet/ MTU Size	Ping test packet size	Comments
ARPA or SLIP	1024	1006	978	Still used with some old Routers supporting ARPA framing (rare)
Ethernet/ DIX or PPP	1518	1500	1472	Preferred size for Ethernet networks
Ethernet/ 802.2 SNAP	1518	1492	1464	Rarely used in Ethernet only networks
Token-Ring/ 802.2 SNAP	1522	1500	1472	Optimized for TCP/IP crossing Eth w/DIX to TR w/Bridges or Routers

Topology/ Frame type	Frame size	IP packet/ MTU Size	Ping test packet size	Comments
Token-Ring/ 802.2 SNAP	1518	1492	1464	Optimized for TCP/IP crossing Eth w/SNAP to TR w/Bridges or Routers (rarely used)
Token-Ring/ 802.2 SNAP	2048	2022	1994	Required for older 4Mb adapters & networks using them.
Wide-Band/ Frame-Relay	N/A	2048	2020	
Token-Ring/ 802.2 SNAP	4096	4070	4042	Optimized for performance
Token-Ring/ 802.2 SNAP	4202	4176	4148	Optimized for TCP/IP with NetWare 3.x/4.x servers
Token-Ring/ 802.2 SNA	4500	4474	4446	Default size for most Routers
FDDI/802.2 SNAP	1542	1500	1472	Optimized for TCP/IP crossing Eth w/DIX to FDDI w/Bridges or Routers
FDDI/802.2 SNAP	1526	1492	1464	Optimized for TCP/IP crossing Eth w/SNAP to FDDI w/Bridges or Routers (rarely used)
FDDI/802.2 SNAP	4096	4054	4026	Optimized for performance
FDDI/802.2 SNAP	4104	4070	4042	Optimized for TCP/IP crossing TR to FDDI w/Bridges or Routers
FDDI/802.2 SNAP	4202	4160	4132	Optimized for TCP/IP with NetWare 3.x/4.x servers
FDDI/802.2 SNAP	4440	4474	4446	Optimized for TCP/IP crossing TR to FDDI w/Bridges or Routers
FDDI/802.2 SNAP	4394	4352	4324	RFC 1188 IETF standard

Topology/ Frame type	Frame size	IP packet/ MTU Size	Ping test packet size	Comments
FDDI/802.2 SNAP	4500	4458	4430	Optimized for TCP/IP using full FDDI packets (not recommended)

Note: Token-Ring allows larger frame/packet sizes than those listed here. In most cases, these are the values used in your network. If you require other values use the formulas in the next section.

VPNs and tunneling encapsulate the normal TCP/IP packet within an outer IP structure. This second IP encapsulation adds additional headers (approximately 20 octets depending on IP options enabled). If the inner TCP data space (MMS) is fully used given the network topology, you can encounter a conflict since there may not be space to apply the additional header and still meet the topologies' frame size limitations. When this happens the VPN device either does not forward the packet or cuts a portion of the TCP data being sent and may not forward this element causing the data to be damaged in transport. You may need to artificially reduce the sending systems MTU size to make sure the TCP data space leaves room for the additional header when it is applied.

Hint: You will need to do both clients and servers since both can send large packets. Consider adding additional NIC to your servers and isolating this traffic.

Refer to your OS documentation for details on altering the MTU or TCP window size as required.

The following formulas are used to derive the values in Table 4-4.

Frame/IP packet sizing

Ethernet with DIX frame size - MAC headers (18) = IP packet size

Ethernet with LLC frame size - MAC headers (18) - LLC/SNAP headers (8) = IP packet size

Token-ring frame size - MAC headers (18) - LLC/SNAP headers (8) = IP packet size

FDDI frame size - MAC headers {ANSI standard} (34) - LLC/SNAP headers (8) = IP packet size

Ping test data size

IP packet - IP headers (30) - UDP (TCP) headers (12) = Ping test packet size (TCP data size)

Note: All measurements are in octets. Bytes is quite often used as the term but technically it is not the same.

Note: We have not allocated any space for either IP or TCP optional fields if they are in use. Tag switching and QOS services quite often add these fields, and their usage space needs to be added to the measures here.

Within a flat network, make sure the settings of the Notes and Domino server systems are set for the largest frame/packet size workable for the LAN topology. If you have a switched, bridged or routed network with either token-ring or FDDI, it is recommended you alter the frame/packet sizing to 4096 to better match the switches', bridges', and routers' memory buffers. With a mixed topology network, you may need to use one of the optimized choices listed in the table above. Here are the guidelines you should follow:

- ▶ Transparent bridge or straight-routed networks with token-ring or FDDI: Use the table entries as needed.
- ▶ With token-ring source route bridged networks: An additional 30 bytes must be subtracted to account for the RIF field in the IP packet and MTU settings. This only applies to token-ring or FDDI network values and ones that are not optimized for Ethernet.
- ▶ Translational bridge between token-ring or FDDI and Ethernet networks: Use the optimized listings that offer the same IP packet size as required (common size).
- ▶ Transparent or source route bridges between FDDI and token-ring: Use the optimized listings as required.
- ▶ ATM with either ClassicIP or LANE should be matched to the same LAN topology the other end node is using.
- ▶ WAN links using ATM should use smaller packet sizes to compensate for the BER of the link.
- ▶ With wide area networks using T1/E1, T3/E3 wide band links or frame-relay links, match them to the LAN topologies MTU or alter the LAN topology's MTU to match wide band or frame-relay link.

Note: With WAN connections, you may want to add a second NIC in the Domino server systems. This offers a way to tune the TCP/IP stack for the constraints of the WAN link without affecting local user or server access.

Because IP is a fragment-capable protocol, networks that have systems located on dissimilar topologies, with a direct router connection offering fragmentation services, do not require alteration of the frame/packet size to meet the requirements of the smaller size allowable between the topologies (maximum common size). If the systems are crossing over a larger frame-capable topology network and the maximum size at each of the end node systems are the same, there is no MTU conflict. Otherwise, with mixed topology networks crossing between switches, bridges (which can't fragment TCP/IP packets), or a router that does not offer fragmentation services, or is disabled, the following changes are required:

- ▶ Between token-ring or FDDI to Ethernet networks, or between token-ring to FDDI networks: Use the optimized listings that offer the same IP packet size as required (common size).
- ▶ Between token-ring or FDDI networks accessing across a leased line, fractional T1/E3, T1/E1, fractional T3/E3, T3/E3 or frame-relay network: Use the optimized listings that offer the same IP packet size as required (common size). With these networks we strongly recommend multi-homing so the local LAN traffic accesses a separate NIC than the remote traffic, which requires the tuned interface.
- ▶ When using GEO stationary satellite up/down link access, you need to multi-home the Domino server with separate NICs at each location so the MTU and TCP window sizes can be manually tuned for the satellite path latency.

If the Router is doing a lot of fragmentation you may want to alter the router's port MTU setting to match the smaller segment's MTU setting (forcing the end node systems to use smaller packets). In the case of a Domino server, you may want to add a second NIC in the system. This offers a way to tune the TCP/IP stack for the constraints of the pathway without affecting other local user or server access.

Your network may require different values if there are network segments set up with lower frame sizes. Use the largest value your network can support.

4.3 RAID

Many administrators are moving to RAID configurations in an effort to increase reliability and performance. This section describes the possible RAID configurations, and the effect of each with respect to Domino.

- ▶ RAID 0 organizes data sequentially across a "stripe" of multiple disk drives. RAID 0 offers high performance at low cost, but since it provides no data protection, it is not recommended for Domino. A single disk failure means you lose all data.

- ▶ RAID 1 is a mirrored set of two physical disk drives, making identical copies of the data on each disk. Usable space is 50 percent of total disk space since one disk is a copy of the other. RAID 1 offers high performance and good protection, but at a high cost. For Domino, RAID 1 is appropriate for:
 - Solaris boot disk
 - Domino transaction logs
 - Domino data directories that fit on a single physical disk
- ▶ RAID 0+1 is a stripe of mirrored disks, a combination of RAID 0 and RAID 1. Usable space is 50 percent of the total disk space. You need an even number of drives for RAID 0+1. It has high performance and good protection, but at a high cost. For Domino, RAID 0+1 is appropriate for Domino data directories that are larger than one drive can hold.
- ▶ RAID 1 Enhanced is similar to RAID 0+1 but can use an odd number of disks. Usable space is 50 percent of the total disk space. It has high performance and good protection, but at a high cost. For Domino, RAID 1 Enhanced is appropriate for Domino data directories that are larger than one drive can hold. (Note: A hardware vendor usually supports RAID 0+1 or RAID 1 Enhanced. The performance of RAID 0+1 and RAID 1 Enhanced should be the same.)
- ▶ RAID 5 is striping data and parity across all members of the RAID set. Parity provides good data protection at a low cost by using one disk's worth of space to protect the data. However, write performance is poor due to additional disk I/O to maintain the parity data. Hardware RAID 5 controllers can significantly improve the write performance over software implementations. In Domino, software RAID 5 is appropriate for databases that have low levels of write activity. Hardware RAID 5 may be appropriate for higher levels of write activity. Note: Domino mail servers have a high level of write activity.

Note: RAID can be implemented in software or using dedicated hardware. With software-based RAID the work is placed on the host processors. Hardware RAID offloads the work to the specialized intelligent controller which often contains cache memory to speed up disk writes.

4.4 Domino settings

In this section we discuss some parameters which can be set to improve the performance of your Domino server. The parameters are grouped by main server function:

- ▶ Common settings for all Domino servers
- ▶ Settings for any mail servers

- Settings for Web clients
- Settings specific to partitions and clusters

We also provide a table identifying server tasks that you might want to disable to free server resources.

Note: Use this information as a starting point, but change only one parameter at a time. Monitor your system both before and after changing a parameter.

Note: These settings may also be placed in the server configuration document, under the Notes.ini tab. The settings can be maintained more easily in this document, which is found in the domain's Domino Directory.

4.4.1 Common settings for all Domino servers

Removing unused server tasks

Many server tasks are started by default. If you don't need a task, don't start it! Some server tasks you may be able to disable are listed in Table 4-5.

Table 4-5 Optional Domino server tasks

Task name	Turn it off, if ...
Router	You are not using the server for electronic mail or workflow.
Replica	No other servers replicate with this one.
Calconn Sched	You are not using the server for calendaring and scheduling.
AMgr	You do not run scheduled agents. Agent Manager is not required for WebQuery Agents.
Collect	Your server does not collect statistics for multiple servers.
Event	You do not monitor events on a server.
Billing	You do not collect billing information.
Cldbdir Clrepl	You do not run your server in a cluster.

Note: Running the AdminP task on a server is necessary if you want the Administration Process to perform administrative tasks.

Notes_SHARED_DPOOLSIZE

The UNIX shell environment variable Notes_SHARED_DPOOLSIZE affects the memory allocation behavior of the Domino server.

Set this Solaris environment variable to 8126464 (8 MB minus 256 KB) in the process that is used to start the Domino server, either in a startup script or the Domino user's environment configuration files (.cshrc or .profile).

Example 4-1 is a sample .profile which sets the Notes_SHARED_DPOOLSIZE setting for users using the Bourne or Korn shell.

Example 4-1 Sample .profile.

```
# @(#)local.profile
PATH=/usr/bin:/usr/ucb:/etc:/bin:/usr/proc/bin:/opt/lotus/bin
export PATH
Notes_SHARED_DPOOLSIZE=8126464
export Notes_SHARED_DPOOLSIZE
NSD_LOGDIR=/lotus/nsd-logs/notes1
export NSD_LOGDIR
NOTESDATA_DIR=/lotus/notes1/notesdata
export NOTESDATA_DIR
```

Mount with noatime

UFS (UNIX file system) volumes maintain the time that each file was accessed. Domino does not use the data maintained by UFS. Domino access time is stored directly in the Domino database. Turn off updates to the file access time by mounting the UFS volume with the noatime mount option. This is done by adding noatime to the data volumes mount point in /etc/vfstab. For example:

```
/dev/dsk/c0t5d0s6 /dev/rdisk/c0t5d0s6 /data0 ufs 1 yes noatime
```

Segmap_percent

If you are seeing high filesystem page out rates (as shown in **vmstat -p**) you may benefit from increasing the value of segmap_percent. This parameter adjusts the percentage of memory that the kernel will map into its address space for file system cache. The default value is 12; on a heavily loaded machine with 4 Gb memory we have seen improvements with values as high as 60. You should experiment with this value, starting with values around 20. On systems with large amounts of physical memory you should increase this value in small increments. This parameter is set by adding the following line to the /etc/system file, for example:

```
set segmap_percent=20
```

maxpgio

If you are using multiple swap devices or are using 10,000 rpm disks, you should allow the page daemon to perform all the I/O it wants. The following setting allows up to 16K I/O operations per second. This parameter is set by adding the following line to the /etc/system file, for example:

```
set maxpgio=16384
```

NSF_DBCache_Maxentries

This Notes.ini setting determines the number of databases that a server can hold in its database cache at one time. The syntax is:

```
NSF_DBCache_Maxentries=value
```

where *value* is the number of databases.

Increasing the database cache size can improve system performance, but requires additional memory.

If the Domino console command SHOW STAT DATABASE displays a cache hitrate lower than approximately 95%, set the NSF_DBCache_Maxentries parameter to slightly more than the actual number of open databases or current users you will support on the server.

Note: It is generally sufficient to use the default value for this parameter. If, however, it is determined that it should be modified, the maximum value this parameter can be set to is 10000.

Server_Max_Concurrent_Trans / Server_Pool_Tasks

If your server supports several thousand simultaneously connected Notes clients, set the Server_Max_Concurrent_Trans in the Notes.ini file to -1 (minus one). In addition, set the Server_Pool_Tasks parameter to 100. This parameter determines the number of threads in the server that respond to requests from Notes clients. The default setting for this value is 80, but when Server_Max_Concurrent_Trans = -1, it is effectively disabled and Server_Pool_Tasks alone sets the actual amount of concurrency in the Domino server. For example:

```
Server_Max_Concurrent_Trans=-1  
Server_Pool_Tasks=100
```

4.4.2 Settings for mail servers

Mailboxes

If the Mail.Waiting statistic reported by the Domino server command SHOW STAT MAIL remains high relative to total mail volume, increase the total number of mailboxes using the following steps:

1. In the Server Configuration Document in the Domino Directory, open the Router/SMTP tab.
2. Open the Basics subtab and set the number of mailboxes to two.

3. Restart the server and verify the setting.
4. If the problem persists, gradually increase the number of mailboxes.

Router Connections

If you see many server-to-server sessions closing with the message, "0 documents read, 0 documents written", and there are few mail messages waiting (as reported by **show stat mail**), you may wish to adjust the following settings:

- ▶ MailMaxThreads (the maximum number of router threads for mail transfer to other servers)
- ▶ MailMaxConcurrentXferThreads (the maximum number of router threads for concurrent transfers to another server)
- ▶ Optionally, MailMaxDeliveryThreads (the maximum number of router threads for local mail delivery)

The default values for these settings are:

- ▶ $3 + (\text{NSF_Buffer_Pool_Size in Megabytes} / 32)$ for MailMaxThreads and MailMaxDeliveryThreads
- ▶ MailMaxThreads/2 for MailMaxConcurrentXferThreads

We have found that the default value for the number of threads handling mail delivery is too large for a heavily loaded Domino server transferring mail to a small number of other partitions or remote servers. We saw performance improvements by using the same number of MailMaxConcurrentXferThreads and MailMaxDeliveryThreads as the number of mailboxes that you have configured for the server, and setting MailMaxThreads to the number of Domino servers that the server is directly connected to. For example, if you have configured 2 mailboxes as described earlier, and you regularly transmit mail to another Domino partition on the same server and one remote Domino server, you can start with a value of 2 for each of the three settings above.

These settings are changed in the server's Configuration Settings Document in the Domino Directory. Open the document, then select the ROUTER/SMTP tab, then the RESTRICTIONS AND CONTROLS sub-tab. For MailMaxDeliveryThreads, select the DELIVERY THREADS tab, and set the desired Maximum Delivery Threads value in this form. For MailMaxThreads and MailMaxConcurrentXferThreads, select the TRANSFER THREADS tab, and set the desired values in the MAXIMUM TRANSFER THREADS and MAXIMUM CONCURRENT TRANSFER THREADS fields in this form. You can confirm these settings by issuing the command **tell router stat** at the Domino server

console. Use these values as a starting point only. Appropriate settings will depend on your server mail topology and the mail patterns of your users. For instance, if there is more local mail delivered on your server than in our example, you may want to increase the number of MailMaxDeliveryThreads.

Closely monitor the MAIL.Waiting value from **show stat mail**, and increase these values gradually if you see mail backing up on the server. It is possible to set these values too high, and increase contention within the router.

MinNewMailPoll

Determines how often workstations can contact the server to see if new mail has arrived for the user. This setting overrides the user's selection in the Mail Setup dialog box. You can increase the mail polling interval if there are a large number of mail users on your server and you want to prevent frequent polling from affecting server performance.

There is no default value for this parameter.

The syntax is:

```
MinNewMailPoll = minutes
```

4.4.3 Settings for Web clients

By default, Domino will execute agents triggered by Web browsers one at a time (serially). Configuring Domino to run agents in parallel may improve your application response time. This can be done via a field on the Server Document, on the Internet Protocols->HTTP tab. The field "Run web agents concurrently?" on the bottom right should be set to "Enabled."

4.4.4 Settings specific to partitions and clusters

PercentAvailSysResources

If you have multiple Domino partitions on a single machine, use the PercentAvailSysResource Notes.ini setting to divide the system's resources among the partitions.

Note: This setting should replace the NSF_Buffer_Pool_Size as well as the NSF_Buffer_Pool_Size_MB from previous releases of Domino.

The value is a percent that allocates the resources each Domino partitioned server will use, or, if needed, constrains Domino servers so other processes like a Web server or tape backup agent can share the system.

For example, if the system is being used to support only two partitioned Domino servers we can give each partition half the resources of the machine by setting this parameter to 50 in each server's Notes.ini file.

```
Server 1 Notes.ini
PercentAvailSysResources=50
```

```
Server 2 Notes.ini
PercentAvailSysResources=50
```

For a more complex example, imagine we want to run two application partitions (app01 and app02), two mail partitions (mail01 and mail02) and a Web server that will take 20% of resources. We also decide that we want to allocate more memory to the two mail partitions than the application partitions. We would set the following values in each partition's Notes.ini:

```
For App01 and App02
PercentAvailSysResources=15
```

```
For Mail01 and Mail02
PercentAvailSysResources=25
```

Each application server takes 15% and each mail server takes 25%, which leaves 20% for the Web server.

Cluster considerations

A Domino cluster links multiple Domino servers together so that they appear as one resource from the client perspective. The cluster functions as a “single” provider of resources, enabling client requests to be processed in a timely manner.

If any given server is unavailable or too busy at the time the request arrives, the cluster transparently passes the request to a server capable of handling the work. For more information on clusters and load balancing see 5.2, “Domino clustering” on page 120

4.5 Summary

In this chapter we have discussed some of the methods for tuning the performance of Domino on a Solaris server.



Domino advanced services

In this chapter we discuss the advanced features of the Domino Enterprise Server (DES). Specifically, we describe the following:

- ▶ Partitioning
- ▶ Clustering
- ▶ Billing

5.1 Domino partitioning

A partitioned Domino server can be thought of as running multiple instances of Domino on the same system. Partitioned servers share the same binary files, and hence, run at the same version; but they have individual data directories that allow for independent configuration of resources.

There are several reasons for implementing partitioned servers:

1. Reduction of operating system and hardware limiting factors. Multiple Domino partitions make more efficient use of server resources than a single server instance, as certain processes can be run in parallel.
2. Segregation of server tasks on a single machine. For example, a mail server and application server can have individually tuned tasks, but run on the same hardware.
3. Segregation of independent user communities. For example, partitioning enables the hosting of multiple Domino domains or Domino-hosted Web sites on a single machine.
4. Independent operation of services and resources. Partitions operate independently and do not have any effect on each other. This allows an administrator the flexibility to treat each partitioned server as though it were a separate system.
5. Reduction in administration effort of the platform. There are fewer systems to administer for your Solaris administration.

5.1.1 Installation

Domino server installations on Solaris require three pieces of information:

1. Program directory
2. Data directory
3. The current UNIX user

The UNIX user (or Username) must be a user recognized by the operating system. This user will be made the owner of the files in the Data directory, as well as the owner of the processes run by that server.

Note: Using a different user for each Domino partitioned server is recommended to facilitate the administration of the different servers.

The default value for the UNIX User and UNIX Group values in the UNIX Install Interface is “notes.”

The Program directory, data directory, and username are provided once with these default values:

- ▶ `program_directory = "/opt/lotus"`
- ▶ `data_directory = "/lotus/notes1/notesdata"`
- ▶ `data_UNIX_user = "notes1"`
- ▶ `data_UNIX_group = "notes1"`

To install additional servers, this information must be copied and the path to the data directory changed.

For complete step-by-step installation instructions, see Chapter 3, "Installing Domino R5 on Sun Solaris" on page 43.

5.1.2 Configuration

It is a good idea to run partitioned servers on multi-processor computers and have at least one processor for each partitioned server that you install on the computer.

For example, running three partitioned servers on a Solaris machine with four processors is a good configuration. This would give you one processor for the platform and one processor for each Domino server.

We also recommend the use of dedicated hard disks for the different Domino data directories. See 2.2.3, "Disk space" on page 14 for more details.

Network configuration

There are two ways to set up partitioned servers, either:

- ▶ Using explicit IP addresses
This means using a separate IP address for each partitioned server and one or more network interface cards (NIC).
- ▶ Using port mapping
This is sharing one IP address and one NIC for all the partitioned servers, but assigning a different TCP port to each partitioned server.

Note: You can use the IP alias mechanism to emulate multiple NIC cards in a single card (multi-netting). To learn how to configure multiple IP addresses on Solaris using the alias mechanism, see 9.7.1, "Network setup" on page 251.

Note: External devices are available that can help you manage your network connection. Some of these devices support technologies such as Network Address Translation (NAT), Port Address Translation (PAT), reverse proxy and load balancing (IP sprayer). This chapter addresses Domino server-based technologies, not these external devices.

Multiple IP addresses

If you want to use multiple IP addresses you must first set up Solaris networking with multiple IP addresses and then configure each Domino partition.

Solaris network setup

Solaris TCP/IP allows multiple physical and multiple logical interfaces. This allows a single machine to be assigned multiple IP addresses, even though it may have only one network interface. Physical network interfaces have names of the form:

`driver-name physical-unit-number`

Logical interfaces have names of the form:

`driver-name physical-unit-number:logical-unit-number`

On a very large Domino server we recommend using multiple physical interfaces instead of, or in addition to, multiple logical interfaces. Logical interfaces may improve network performance for traffic between the partitioned servers that share a single physical interface, but it may decrease overall network performance under heavy client load. This depends very much on your environment. Combinations of multiple public and private networks may be required for optimal performance in your network.

You have to configure the network setup of your Solaris box before you can configure Domino servers.

For Domino partitioned servers you have to add additional IP addresses to be used for each partition. First create “/etc/hostname.device:n” files that contain the name of each partitioned server, where device is the device name of the physical interface or logical interface. As described above a physical interface could have the name similar to hme0 while a logical interface would have a name of hme0:1.

Note: In Solaris 7 and 8 the device name usually is hme0, for the first network card, hme1, and so on.

For example, let's consider that you want to create three Domino partitioned servers, where the first two servers share the first physical interface and the third server exclusively uses the second physical interface, as shown in Figure 5-1.

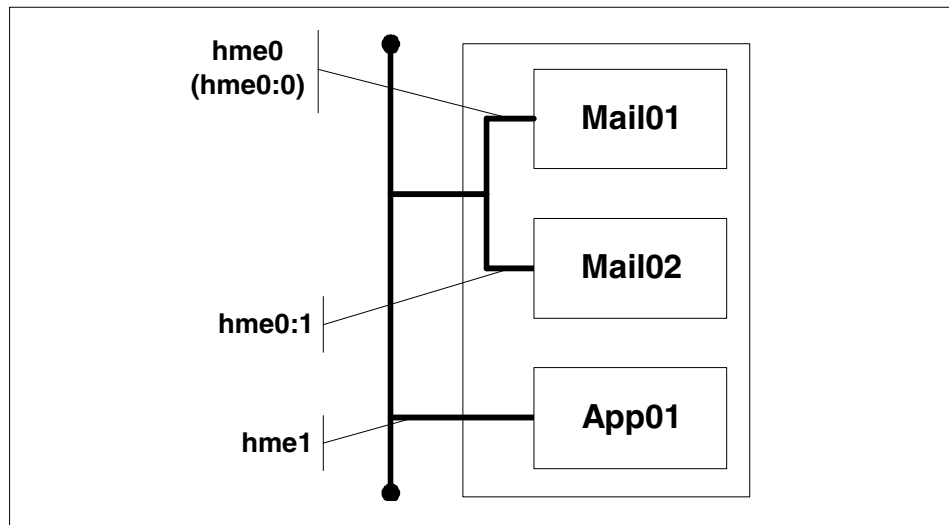


Figure 5-1 Sharing network interfaces

To implement this, the following steps are necessary:

1. The file `/etc/hostname.hme0` file should already exist for the hostname of the physical server and contains the name of the server, in this case `mail01`, which is the first logical interface on the first physical interface.
2. Create the file `/etc/hostname.hme0:1` containing the server name `mail02`, which is the second logical interface on the first physical interface.
3. Create the file `/etc/hostname.hme1` containing the server name `app01`, which is the server running on the second physical interface.
4. Then issue the following commands if you want to manually start the IP interfaces without rebooting the machine. The RC scripts will automatically start and stop your interfaces as you bring the system up and down:

```
# /sbin/ifconfig hme0 plumb (only if hme0 is not already 'plumbed')
# /sbin/ifconfig hme0 <IP_address> up (if hme0 has not been started)
# /sbin/ifconfig hme0:1 <IP_address> up
# /sbin/ifconfig hme1 <IP_address> up
```

where `n` corresponds to the number of the `/etc/hostname.hme0:n` files.

For example: `/sbin/ifconfig hme0:1 10.3.187.210 up`

Note: If you want to disable the IP address, type the command:

```
/sbin/ifconfig hme0:n down
```

If you don't use Domain Name System (DNS), you have to put the new IP addresses in the `/etc/hosts` file. For example:

```
9.95.35.68 Mail01  
9.95.35.53 Mail02  
9.95.35.69 App01
```

This example uses IP addresses from a single network. It is possible to configure this on more than one network. However, logical network interfaces should be on the same network.

If you use DNS, ask your DNS administrator to add new IP hostname aliases into the DNS. In order to use DNS you must configure the Solaris server to be a DNS client and set the order in which you need to resolve host names between the host files and the DNS.

Note: Use DNS instead of host files whenever possible!

Configuring your Notes.ini for partitioned servers

Each Notes server has a unique IP address assigned to it in either the DNS system or the `/etc/hosts` file on the local machine.

In the following example all four servers are using port number 1352, which is the Domino default port number. Each server needs only one line in its Notes.ini corresponding to the server name.

- ▶ Server1
TCPIP_TcpIpAddress=0,192.94.222.170:1352
- ▶ Server2
TCPIP_TcpIpAddress=0,192.94.222.171:1352
- ▶ Server3
TCPIP_TcpIpAddress=0,192.94.222.172:1352
- ▶ Server4
TCPIP_TcpIpAddress=0,192.94.222.173:1352

For more information on port binding see 3.4.1, "IP address binding" on page 74.

Using Port Mapping

To set up partitioned servers using one IP address on one IP interface you can use one port mapper server.

Note: The port mapper server becomes a very critical break point. We recommend that the port mapper is not doing anything other than the port mapping function. This limits the likelihood of the risk or redirection failing.

Port mapping is designed for platforms that do not support the use of multiple IP addresses on one system. However, there are other situations (for example, limited availability of IP addresses) where port mapping can be an option.

Example 5-1 shows a configuration for three partitioned servers using port mapping.

Server1 will be the port mapper server for server2 and server3.

Example 5-1 Port mapping example

Server1

```
TCPIP_TcpIpAddress=0,192.94.222.179:1352
TCPIP_PortMapping00=CN=Server1/O=Org1,192.94.222.179:1352
TCPIP_PortMapping00=CN=Server2/O=Org1,192.94.222.179:13520
TCPIP_PortMapping01=CN=Server3/O=Org1,192.94.222.179:13521
```

Server2

```
TCPIP_TcpIpAddress=0,192.94.222.179:13520
```

Server3

```
TCPIP_TcpIpAddress=0,192.94.222.179:13521
```

Each server's name must be associated in the DNS as CNAME records (server2 & server3) to the A record of the port mapping server (server1). If using Host files on the remote systems, you will need to multi-list (alias) each port-mapped Domino server's name with the same IP address.

Within a connection document, in addition to placing the numeric IP address, you can also append it with the given server's TCP port (i.e. 13520). This by-passes the port mapper server. For example, using 192.94.222.179:13520 in the Network Address field of a connection document from either Server1 or a Notes client would connect to Server2 directly. For more information see Appendix E, "Example TCP port Notes.ini settings" on page 385.

5.1.3 Configuring memory resources for partitioning

Some tuning recommendations are appropriate for a Domino partitioned environment.

Note: Refer to the previous chapter for more information on tuning.

PercentAvailSysResources setting in Notes.ini

The PrecentAvailSysResource setting is now the preferred means to partition the system resources a given Domino server will use. This setting should replace the NSF_Buffer_Pool_Size as well as the NSF_Buffer_Pool_Size_MB.

In R4.xx and earlier versions of R5.0x, Lotus depended on the NSF_Buffer_Pool_Size setting to enlarge in the case of R4.xx, and limit in the case of R5.0x, the Domino server's use of RAM memory for NSF buffers. While this did what it was intended, it did not address the other buffer and handle tables Domino uses in R5.0x. This is why Lotus introduced the newer setting PrecentAvailSysResource. This parameter sets the limit the given server's NSF buffer will be allowed to grow. It also affects other settings in the correct proportion needed for the available memory the given OS has free. It is using a percent value, so it is easier to allocate the resources a Domino partitioned server will use; or if needed, constrain the Domino servers so other processes like a Web server or tape backup agent can share the system and have the needed resources they require.

For example, imagine that we want to run two application partitions (app01 and app02), two mail partitions (mail01 and mail02) and a Web server that will take 20% of memory. We also decide that we want to allocate more memory to the two mail partitions than the application partitions. We would set the following settings in each partition's Notes.ini:

For App01 and App02

```
PercentAvailSysResources=15
```

For Mail01 and Mail02

```
PercentAvailSysResources=25
```

Each application server takes 15% and each mail server takes 25%, which leaves 20% for the Web server.

Your requirements will dictate what resources to allocate to each of your servers.

The values used must be whole numbers. If your percent is not a whole number it's best to round down. Some server functions don't require large memory allocations, such as a Domino server which is exclusively used as an SMTP gateway or as a Notes pass-through server. In addition, you may have other applications which require physical RAM, so the available memory we want the Domino servers to use might be different.

Solaris systems use memory mapped files (mmap), and therefore, they create tmp files within the swap partition (/tmp). The space allocation of the /tmp volume must be at least the same as physical memory, otherwise you can run out of disk space. Virtual memory is not involved in the memory sizing algorithm, but is used by the Domino server for shared memory segments. Other UNIX OSes use SystemV interprocess communication facility (ICP); thus they don't rely on swap space.

Attention: Previous releases of Domino R5 used the `NSF_Buffer_Pool_Size` setting for partitioned servers, which has since been replaced by the `PercentAvailSysResources` setting. Entries to the old `NSF_Buffer_Pool_Size` setting should now be replaced with the new `PercentAvailSysResources` setting.

Notes shared memory

Starting with Release 5, Domino no longer uses the IPC shared memory on Solaris. Shared *memory mapped files* are used instead.

Starting from release 5, the default size of the memory mapped files is 8 MB. For a Domino environment on Solaris this value is too large, and wastes virtual memory.

The `Notes_SHARED_DPOOLSIZE` environment variable is used to modify the size of the shared memory segments. The value of this variable is expressed in bytes. For Domino on Solaris, this variable should be set to 8126464, in the environment of the user who runs the Domino server.

For example, if the user runs the C shell:

```
setenv Notes_SHARED_DPOOLSIZE 8126464
```

If the user runs Korn or Bourne shells:

```
Notes_SHARED_DPOOLSIZE=8126464
export Notes_SHARED_DPOOLSIZE
```

Each Domino server environment will have different levels of memory usage. The Domino administrator should monitor the number of memory mapped files created in /tmp, and the amount of free space in /tmp.

5.1.4 Troubleshooting

Troubleshooting a Domino partitioned server is quite similar to the process on a standard Domino server.

The only thing to know is that each Solaris command you use to monitor the Domino server should be filtered using the **grep** command with the Domino user.

For example, if you want to check all the processes for the Domino server running as the notes2 user, use the command:

```
ps -ef | grep notes2
```

If a partitioned server encounters a fatal error, you can restart that partitioned server without restarting the computer. It is best to use different user accounts (log-in names) for each partitioned server so you can use the appropriate command, such as **nsd -kill**, to clean up residual processes after a server crash.

How to selectively kill one of the partitions

1. Create a separate administrator user, a UNIX account dedicated to running one Domino server partition, for each of the partition servers, for example notes1, notes2, notes3. Be sure that they are all created in the “notes” group.

Note: Ideally, these users would be created before Domino was installed, and files in each Domino server partition would be owned by the appropriate administrator user.

2. Launch each partitioned server using each separate administrator user. Because the users are members of the “notes” group, the server will have the necessary privileges to run.
3. Should you need to kill one of the partitioned servers, navigate to the data directory for that partition, and as that server partition's administrator user execute the following command:

```
/opt/lotus/bin/nsd -kill
```

This will kill all processes and threads started by this user, effectively killing one partition.

5.2 Domino clustering

A Domino cluster links multiple Domino servers together so that they appear as one resource from the client perspective. The cluster functions as a “single” provider of resources, enabling client requests to be processed in a timely manner.

If any given server is unavailable or too busy at the time the request arrives, the cluster transparently passes the request to a server capable of handling the work.

The cluster members can be on a mixture of the supported Domino platforms, including Windows NT, various UNIX systems, IBM AS/400, and OS/2. The clusters support Notes clients only. For Web browsers you need a different solution, like ICM. (See 9.8, “The Internet Cluster Manager (ICM)” on page 258 for more information.)

Domino clustering is accomplished entirely at the application level. No special hardware is needed, but be aware that you need to add resources to your system (CPU and RAM). Clustering has to be considered in the sizing of a system.

Domino clusters replicate database changes to all replica copies of the database as the changes occur. This synchronization of cluster components is key to Domino’s high availability. This style of replication is referred to as event-driven (immediate) replication, in contrast to standard replication that occurs on a schedule. Event-driven replication is a function of the cluster replicator.

5.2.1 Workload balancing

With clustering, multiple copies of databases on multiple servers provide high availability. In addition, Domino distributes the workload between the cluster members (this is called workload balancing), allowing for lower overall response times and more consistency in response times during peak intervals.

Domino clusters provide workload balancing by redistributing user requests to an overloaded server to other servers in the cluster that have available capacity. To optimize workload balancing, you can modify the following settings:

- ▶ Database distribution
- ▶ Server availability
- ▶ Specify different home servers in the location document of the clients

Database distribution

Make sure that you distribute databases evenly in the cluster. When a server in the cluster fails or becomes overloaded, user requests automatically redirect to other servers in the cluster.

Ideally, this load should be spread equally across all other servers in the cluster. However, this can only happen when replicas of the databases on the failed server are spread roughly equally across the other servers in the cluster.

Note that if you distribute the databases evenly across the servers, you're assuming that the databases have about the same activity. If you have some power users or particularly active databases, you may need to fine tune the distribution of those databases to make the activity on each server approximately equal.

Server availability

Set the threshold for when the server is considered "Busy." Each server in a cluster periodically determines its own workload, based on the average response time of requests recently processed by the server.

The server availability index indicates how busy the server is. The index is a value between 0 and 100, where 100 indicates a lightly loaded server (fast response times), and 0 is a heavily loaded server (slow response times).

With the Notes.ini variable `SERVER_AVAILABILITY_THRESHOLD`, you can specify a threshold that determines the lowest value of the server's availability index for which the server is not considered busy. When the server's availability index goes below the threshold value, the server is in the busy state. A server in the busy state redirects users to another server in the cluster.

The server's availability index is derived from the ratio between the current response time and the response time in optimum conditions (with no Domino transactions). Note that the response times that are taken into account are server-based and do not include any consideration for network time. The Cluster Manager process on each server monitors the average response time of a set of server operations over roughly the last 75 seconds.

Domino uses the Notes.ini setting `SERVER_TRANSINFO_NORMALIZE` when calculating the server availability index to "normalize" the response times observed at the server (that is, it divides the observed response times by this normalize value).

For the availability index calculation to work properly, the normalize value should be roughly equal to the average Domino transaction time (for the server in question) in milliseconds* 100.

The default value is 3000 ms, corresponding to an average response time of 30 ms per transaction. This default setting was appropriate for "the average server" when clustering was first shipped several years ago, but may be considered too large for the current generation of servers. You could try using a lower normalize value with today's faster servers, to improve failover times. For more information on tuning this parameter, see the article *Optimizing Server Performance: Domino Clusters*, available at <http://www.notes.net>.

5.2.2 Failover

In a Domino cluster, if one member of the cluster fails, another member of the cluster transparently assumes the failing member's workload.

This action is called failover. Failover is client-driven, and when a client cannot connect to a server it checks in its cluster cache for other servers in the cluster and thus redirecting requests to another server in the cluster that has a replica of the database needed to service the request.

Tip: To quickly test a failover use the `SERVER_RESTRICTED=1` variable from the server console "set config `SERVER_RESTRICTED=1`." A server in restricted state does not accept any connection. Remember to disable the server restriction by resetting the variable to 0.

5.2.3 Creating the cluster

Using the administration client, open the Domino Directory database on the server specified as the Administration server for the domain.

Use the following steps to create the cluster:

1. Click the Configuration tab.
2. Expand the Server section in the left panel.
3. Click the All Server Documents view.
4. Select the servers that will be in the cluster that you are about to create.
5. Click the Add to Cluster action button, as shown in Figure 5-2.

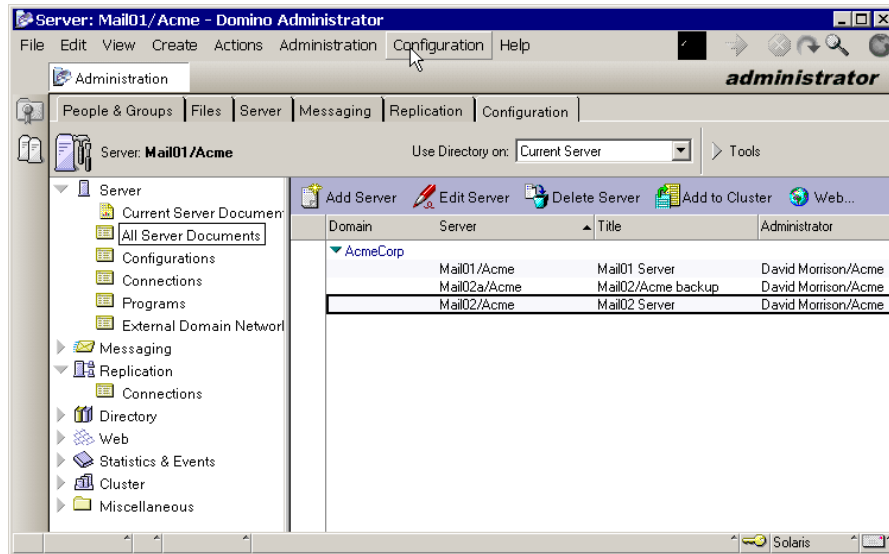


Figure 5-2 Adding a Domino server to a cluster

- Confirm that the action should continue by clicking Yes in the Verification dialog box.

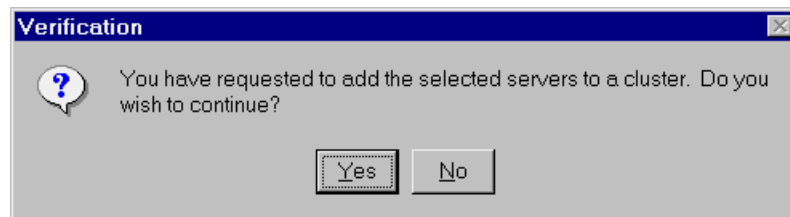


Figure 5-3 Adding a new server confirmation dialog

- For a new cluster, choose the Create New Cluster option and click OK, or choose the appropriate existing cluster name.

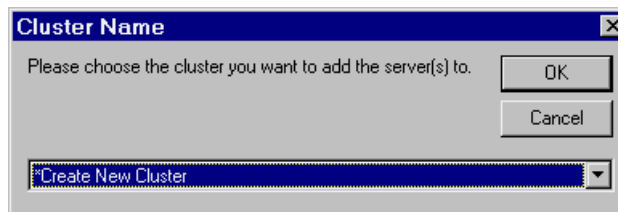


Figure 5-4 Selecting the new cluster

8. Type in the name of the cluster and click OK.

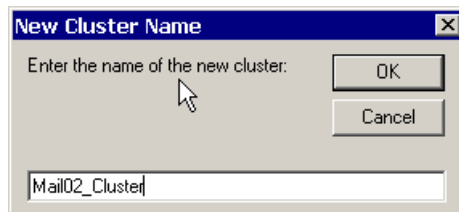


Figure 5-5 Adding a new server cluster name

9. Decide if the cluster should be created immediately or through the administration process and click the appropriate button. If your system allows it, always choose "No" to let the administration process perform all required steps.

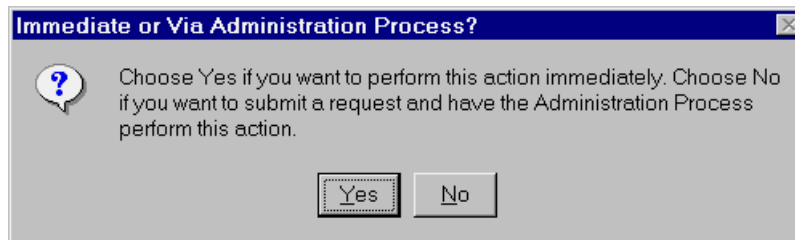


Figure 5-6 Choosing when to submit the new cluster request

10. If you chose Yes in step 9, a confirmation dialog box will be displayed.



Figure 5-7 New cluster confirmation dialog box

11. To see your cluster configuration, expand the Cluster section in the left panel and choose the Clusters view, as shown in Figure 5-8.

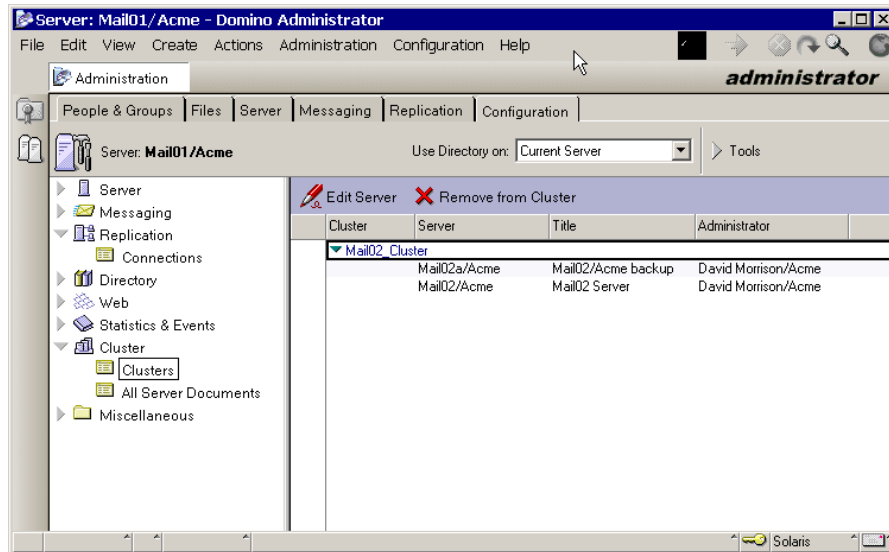


Figure 5-8 Viewing your new cluster in the Administrator client

12. In the server document the field Cluster Name will be filled with the name of the cluster, as shown in Figure 5-9.

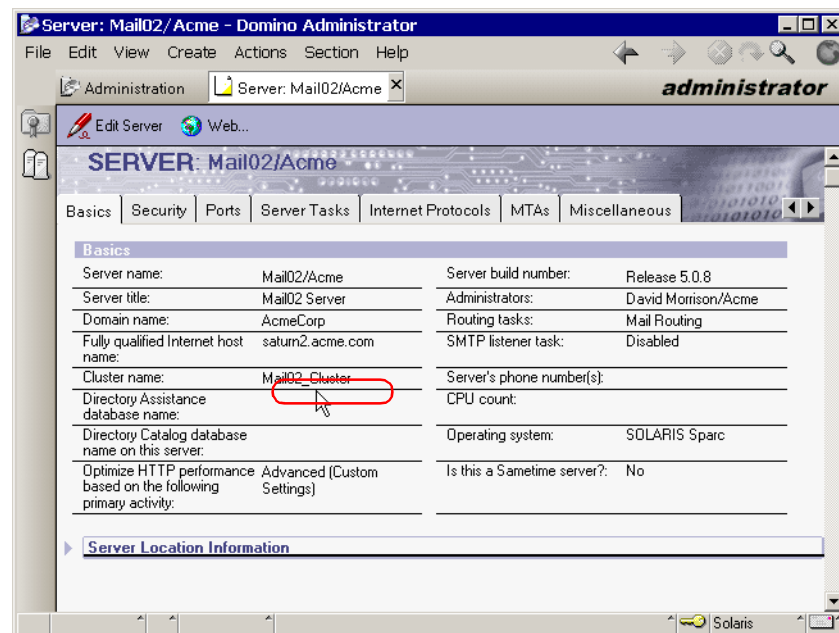


Figure 5-9 Displaying a servers cluster name in the server document

5.2.4 Cluster directory database

The cluster directory database, clbdir.nsf, contains all the databases that you want replicated between the cluster members.

It is created automatically when the cluster is configured and it contains all the databases and templates located in the Domino data directory.

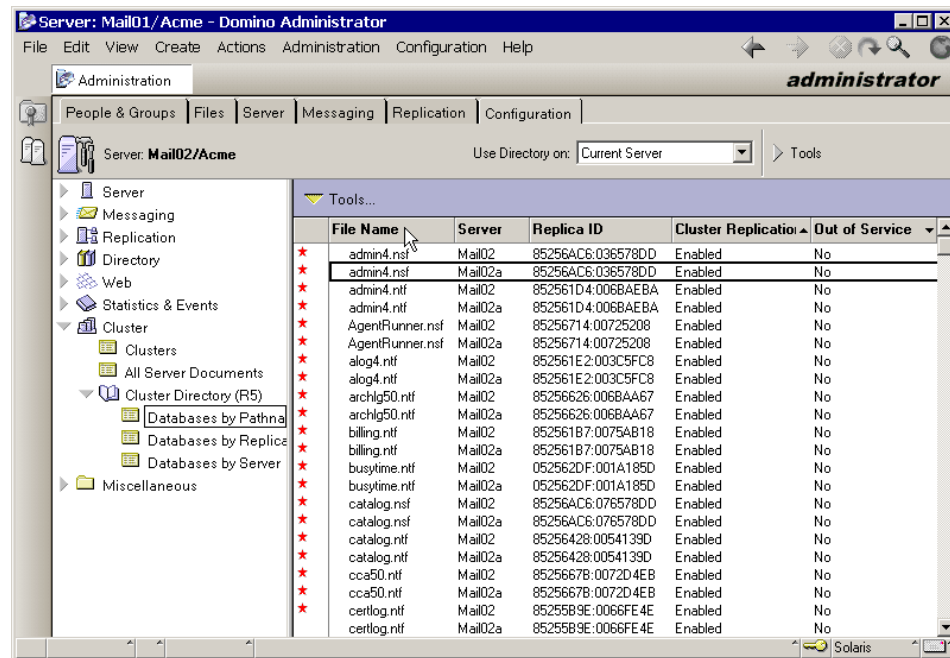


Figure 5-10 Viewing the Cluster Directory database

Click Tools - Disable Cluster Replication of Selected Databases to disable cluster replication of non-critical databases and templates.

To re-enable them click Tools - Enable Cluster Replication of Selected Databases.

5.2.5 Removing a server from a cluster

Using the administration client, open the Domino Directory database on the server specified as the Administration server for the domain.

Use the following steps to remove a server from a cluster.

1. Click the Configuration tab.

2. Expand the Cluster section in the left panel.
3. Click Clusters.
4. Select the servers that will be removed from the cluster.
5. Click the Remove from Cluster action button.

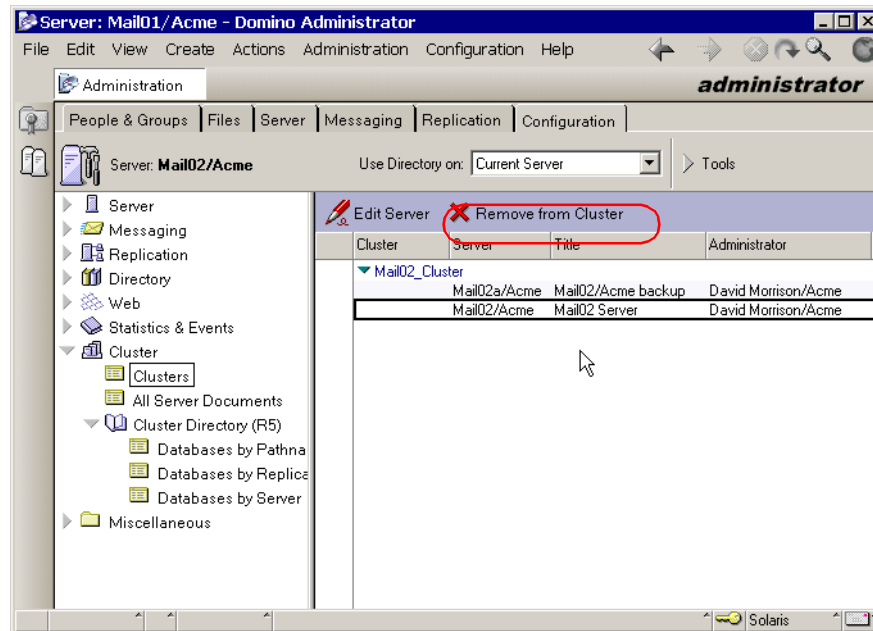


Figure 5-11 Selecting a server to be removed from a cluster

6. Confirm that the action should continue by clicking Yes in the Verification dialog box.

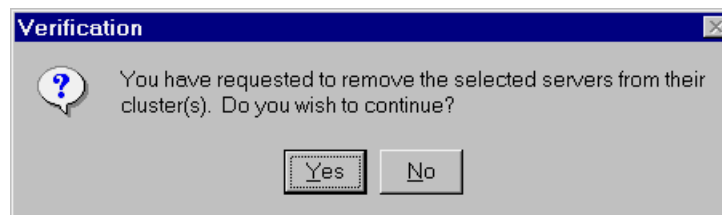


Figure 5-12 Confirming the removal of a server from a cluster

7. Decide if the server should be removed immediately or through the administration process and click the appropriate button, as shown in Figure 5-13 on page 129.

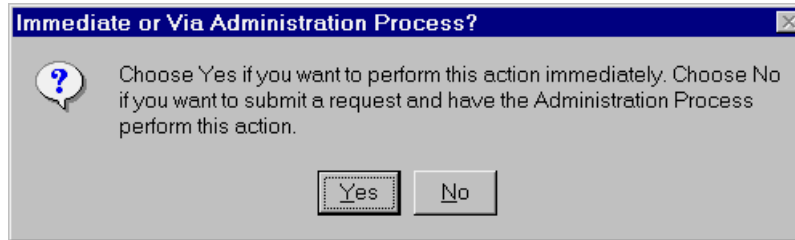


Figure 5-13 Selecting when to remove the server from a cluster

8. If the choice in step 7 was Yes, a confirmation dialog is displayed.



Figure 5-14 Immediate confirmation of a servers removal from a cluster

9. If the choice in step 7 was No, the request is submitted to the administration process.

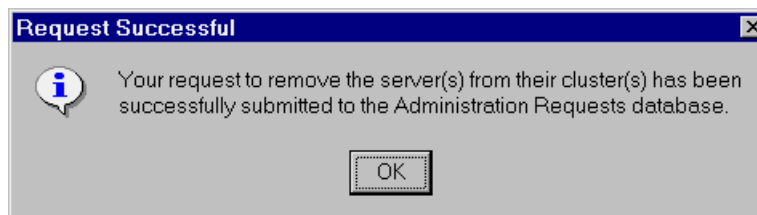


Figure 5-15 Server removal from a cluster confirmation from AdminP

10. The cluster processes clrepl and clbdir are stopped on the server, and the entries removed from the ServerTasks= line in Notes.ini.
11. The database entries in the Cluster Directory database (clbdir.nsf), for databases on the server being removed, are purged from the replica copy on the least busy of the remaining cluster servers.
12. The local Cluster Directory database is deleted from the server that is being removed from the cluster.

5.2.6 Setting up your cluster

When setting up your cluster, you should consider the benefits of using a private LAN for intra-cluster communication. This way, you can offload the cluster's probe and replication network traffic from the LAN, leaving more bandwidth for client communication with the cluster servers. You can also eliminate the network as a single point of failure in your cluster.

Tip: You can direct all server-to-server communication within your cluster through the private LAN. In the Ports= line in the Notes.ini of each cluster server, place the cluster port at the first position.

How many servers in a cluster

Another basic decision when setting up a cluster is how many servers to put in the cluster. If you are just getting started with Domino clustering, you'll probably want to start with just two or three. But, as your confidence grows, you may want to add more servers. Lotus states that a cluster can have as many as six servers.

While it's true that the overhead of cluster probes and cluster replication could begin to degrade performance when there are more than six servers in the cluster, there is no hard limit imposed by Domino.

How many CLREPL tasks

A useful guideline for determining the number of cluster replicators to run on each server is to set up as many cluster replicators as there are cluster members, minus one. For example, if there are four servers in the cluster, enable three CLREPL tasks on each server in the cluster. In this way, there will always be at least one CLREPL task available to handle replication to each server in the cluster.

Another way to determine how many cluster replicators to run is to monitor the cluster replication statistics.

Replica.Cluster.WorkQueueDepth shows the number of modified databases that are waiting to be replicated. If the number of databases waiting is frequently greater than zero, you should consider adding one or more cluster replicators.

However, this number may also be greater than zero if you don't have enough network bandwidth to process the transactions fast enough. If this is the case, you should consider setting up a private LAN for your cluster.

Also keep in mind that adding processors or memory to your server can enhance replication performance.

Replica.Cluster.SecondsOnQueue shows the number of seconds that the last database spent waiting in the replication queue before being replicated. Since the cluster replicator checks its queue every 15 seconds, this number should be under 15 during periods of light load. If this number is frequently higher than 30, you should consider adding one or more cluster replicators.

Enabling multiple cluster replicators

Use one of the following procedures to enable multiple cluster replicators:

- ▶ Starting multiple tasks at server startup
Append one or more CLREPL tasks to the ServerTasks setting. For example:
ServerTasks=Router,Update,Adminp,CLDBDIR,CLREPL,CLREPL,CLREPL
- ▶ Starting multiple tasks manually
To manually enable multiple Cluster Replicators for the active server session, type the following at the server console for each cluster replicator you want to start:
load clrepl
Each time you enter this command, the server starts another cluster replicator task.

Note: Be sure all CLREPL tasks are after CLDBDIR in the ServerTasks setting.

5.2.7 Cluster statistics

To obtain cluster statistics, execute the command shown in Example 5-2 at the Domino console.

Example 5-2 Displaying cluster statistics

```
> show stat server.cluster.*
Server.Cluster.Member.Mail02a/Acme.Index = 100
Server.Cluster.Member.Mail02/Acme.Index = 100
Server.Cluster.Name = Mail02_Cluster
Server.Cluster.OpenRedirects.FailoverByPath.Successful = 0
Server.Cluster.OpenRedirects.FailoverByPath.Unsuccessful = 0
Server.Cluster.OpenRedirects.Failover.Successful = 0
Server.Cluster.OpenRedirects.Failover.Unsuccessful = 0
Server.Cluster.OpenRedirects.LoadBalanceByPath.Successful = 0
Server.Cluster.OpenRedirects.LoadBalanceByPath.Unsuccessful = 0
Server.Cluster.OpenRedirects.LoadBalance.Successful = 0
Server.Cluster.OpenRedirects.LoadBalance.Unsuccessful = 0
Server.Cluster.OpenRequest.ClusterBusy = 0
Server.Cluster.OpenRequest.DatabaseOutOfService = 0
Server.Cluster.OpenRequest.LoadBalanced = 0
```

```
Server.Cluster.PortName = *
Server.Cluster.ProbeCount = 108
Server.Cluster.ProbeError = 0
Server.Cluster.ProbeTimeout(mins) = 1
>
```

To get the statistics relative to replication use the command shown in Example 5-3.

Example 5-3 Displaying replication statistics for a cluster

```
show stat replica.cluster.*
Replica.Cluster.Docs.Added = 46
Replica.Cluster.Docs.Deleted = 2
Replica.Cluster.Docs.Updated = 4
Replica.Cluster.Files.Local = 80
Replica.Cluster.Files.Remote = 80
Replica.Cluster.Retry.Waiting = 0
Replica.Cluster.SecondsOnQueue = 9
Replica.Cluster.SecondsOnQueue.Avg = 8
Replica.Cluster.SecondsOnQueue.Max = 14
Replica.Cluster.Servers = 1
Replica.Cluster.SessionBytes.In = 8602
Replica.Cluster.SessionBytes.Out = 82772
Replica.Cluster.Successful = 8
Replica.Cluster.WorkQueueDepth = 0
Replica.Cluster.WorkQueueDepth.Avg = 0
Replica.Cluster.WorkQueueDepth.Max = 2
>
```

Check these statistics regularly to monitor your cluster configuration. For details about the meaning of these statistics, see the “Cluster statistics” section in the Domino 5 Administration on-line help database.

5.2.8 Troubleshooting

Among the items to check if you encounter problems with your cluster configuration are those discussed in this section.

Check the Notes.ini

For correct cluster configuration, look at the ServerTasks variable and the Server_Cluster_On variable.

The ServerTasks line should include the tasks CLDBDIR and CLREPL. CLDBDIR is responsible for building and maintaining the cluster database directory, CLDBDIR.NSF. The cluster database directory must exist for the cluster replicator to load.

If the variable Server_Cluster_On=1 is set, the cluster manager will be enabled; otherwise, the cluster manager is disabled.

The variable Server_Cluster_Default_Port assigns what port the cluster replicator uses to PUSH changes to cluster members. If cluster replication isn't pushing changes, this variable may be set to a non-existing port. If this variable isn't in your Notes.ini, the cluster replicator will use one of the enabled ports.

Logging

To enable logging of cluster replication events, enter the following in the Notes.ini file:

```
RTR_logging=n
```

where the possible values for n are:

- ▶ 1 = enable logging of cluster replication events
- ▶ 0 = disable logging of replication events

Debug

Use the variable SERVER_DEBUG_CLUSTERS=1 in the Notes.ini file to have more information on the cluster activity.

Cluster database cache

Since the cluster replicator may receive several modification events for one database in a relatively short period of time, it is designed to keep a cache of open databases so that databases are not closed and reopened frequently during periods of high activity.

The cache allows up to 64 local databases (and each of their replicas on other servers) to be kept open for up to 10 minutes in anticipation of further modifications needing replication to these databases.

You should be aware of this cache if you need exclusive access to a database on one of the servers. The cache can be disabled by setting the configuration parameter, RTR_Cached_Handle_Disable, to a non-zero value in the Notes.ini file.

The number of local databases currently open by the cluster replicator can be seen by examining the Notes statistic Replica.Cluster.CachedHandles.

5.3 Billing

Use billing to collect information about the activity in your enterprise. Then, using the information that billing collects, you can charge users for the amount that they use your system, monitor usage trends, conduct resource planning, and determine if clustering would improve the efficiency of your system.

5.3.1 How it works

When you enable billing, Domino collects information about client and server activity and places this information in the billing message queue. Periodically, the billing task polls the message queue and moves the billing information to a destination that you specify—a Notes database, a binary file, or both.

To create billing reports, you write an application to access the billing information. If you collect the information in a Notes database, you can write a Notes API program to create the billing reports you want. If you collect the information in a binary file, use a third-party program to analyze the data and create billing reports.

5.3.2 Configuration

Edit the Notes.ini file to set up billing. Within the Notes.ini file, you enter commands to:

- ▶ Start the Billing task
- ▶ Specify which billing classes to track
- ▶ Specify where to store billing records

Starting

Edit the ServerTasks setting in the Notes.ini file to include the Billing task. For example, the line might read:

```
ServerTasks=Replica,Router,Update,Stats,Billing
```

Billing classes

Add this line to the Notes.ini file to specify which billing classes to track:

```
BillingClass=list
```

where list contains one or more of the following:

- ▶ Agent - Tracks the user, task, and database activity related to agent execution on the billing server.

- ▶ Database - Tracks each database opened and closed on the billing server. Database billing tracks the amount of elapsed time a database is open per session.
- ▶ Document - Tracks access for documents in a database that contain hidden BILLCHARGERED or BILLCHARGEWRITE fields and associates a cost charge with the action.
- ▶ HTTP Request - Tracks Web server requests.
- ▶ Mail - Tracks the mailing of documents. When you enable mail billing, mail messages leaving the billing server mailbox are tracked and recorded in mail billing records.
- ▶ Replication - Tracks replication of databases when initiated by the billing server with another Lotus Domino server.
- ▶ Session - Tracks network traffic that a user generates during a session. For example, when the user logs on to the server, Domino records any network-related activity needed to complete the transaction.

Storing the information

Add this line to the Notes.ini file to specify where to store billing information:

```
BillingAddinOutput=n
```

where n is one of the following:

- ▶ 1 to store the records in a Notes database
- ▶ 2 to display the records on the server console
- ▶ 8 to store the records in a binary file
- ▶ 9 to store the records in both a database and a binary file

Tuning

By default, the Billing task starts running once every minute and runs for 10 seconds each time. To adjust these default behaviors, edit these settings in the Notes.ini file:

- ▶ BillingAddinWakeup=seconds, specifies how often the billing add-in task runs.
- ▶ BillingAddinRuntime=seconds, specifies how long the billing add-in task runs.
- ▶ BillingSuppressTime=minutes, specifies the frequency of record stamping during session and database activities. If you want billing data collected more frequently, shorten the default value of 15 minutes. To minimize the billing workload on your system, lengthen the value.

Note: The BillingAddinWakeup value must be greater than the value you specify for BillingAddinRuntime.

5.3.3 Troubleshooting

You can run multiple billing tasks to increase performance. Do this by adding the billing task in the Notes.ini, ServerTask line, or by loading the task from the Domino console:

```
> load billing
```

Domino creates the BILLING.NSF database the first time it runs the billing task. If you experience problems, check that the variable BillingAddinOutput=n is present in the Notes.ini file of the server.

5.3.4 Customizing billing

The billing message queue is the heart of the billing process. All billing messages generated by the billing server are written to this queue. In turn, the billing server task retrieves the messages from the queue for further processing.

The Lotus C API for Domino and Notes provides functions for reading from and writing to the billing message queue, and also defines the data types and symbols that comprise the contents of the message.

You can make your own billing class to collect all the information that you want.

The Lotus C API documentation contains a good billing sample BILLSES to illustrate how to program a billing add-in. BILLSES is loaded as a Lotus Domino server task. It periodically reads billing messages off the message queue, filters out all session and database billing class records, and appends the information to a predesigned database.

The sample reads both standard and custom (extended) billing messages generated on the Lotus Domino server. The sample is shipped with a Makefile for AIX. We compiled it on Solaris using the GNU gcc compiler version 2.95.2. A similar Makefile is in Appendix A, "Using Notes C API to make your own backup tool" on page 363

5.4 Summary

In this chapter we described the advanced services in Domino R5. We focused on the installation, configuration, and troubleshooting procedures for partitioning, clustering, and billing.



Administration

In this chapter we describe the different ways you can access your Domino server to do administration. Because there is no Notes client for the Solaris platform, you must administer your Domino servers with one of the administration tools described in this chapter.

You'll get an overview of:

- ▶ The Domino Administrator client, which provides you with tools for graphical monitoring of servers, services, replication, and routing. You can use the Domino Administrator client to perform most administration tasks.
- ▶ The Web Administrator, which allows you to manage and view settings for a Domino server with a browser. It helps you to keep your Domino server up and running, even if you don't have access to a Domino Administrator client.
- ▶ The Domino Character Console (the cconsole program), which provides a way to access the server console from the UNIX command line. This feature is supported only for UNIX platforms. (Domino for AS/400 has something very similar.)

6.1 The Domino Administrator client

Previous versions of Lotus Domino had a single, all-purpose Notes client that would be used by users, administrators, and application developers.

As a result of the strong focus on ease of use in the design of Lotus Domino R5.0, three individual clients are now available. They are:

- ▶ Notes R5.0, the user's client
- ▶ Domino Administrator R5.0, the administrator's client
- ▶ Domino Designer R5.0, the developer's client

In this chapter, we take a look at the Domino Administrator client and some of its features. If you want to learn about this tool in more detail, refer to the Domino online help system and the product documentation that ships with the Domino R5 server.

The Domino Administrator client is available for Win32 only.

6.1.1 Overview

The Domino Administrator client has a look and feel similar to that of the standard Notes client. The interface is intuitive and task-oriented. You can administer all Domino servers in your enterprise from the same workstation, even if they are in multiple Domino domains.

To leverage your existing skills and experience, the Domino Administrator client implements common features such as:

- ▶ Drag and drop
- ▶ Multiple selections using the Shift and Ctrl keys
- ▶ Right-click context-sensitive menus

Important new server monitoring features allow you to monitor and manage your environment proactively. Finally, you now have the ability to configure, manage, and enforce user desktop settings centrally. All these administration enhancements, and more, result in the most comprehensive server management tool and reduce the cost of ownership.

6.1.2 Starting the Domino Administrator client

There are at least three ways you can start the Domino Administrator client:

- ▶ Double-click the Domino Administrator icon on your desktop.

- In the Notes client, click the Domino Administrator bookmark button.
- In the Notes client, choose **File -> Tools -> Server Administration**.

After you start the Domino Administrator, an Administration window appears. This window has three main areas: the server list, tabs, and tools.

6.1.3 Using server lists

The first time you start the Domino Administrator client, the system automatically creates a server list similar to the one shown in Figure 6-1 on page 139.

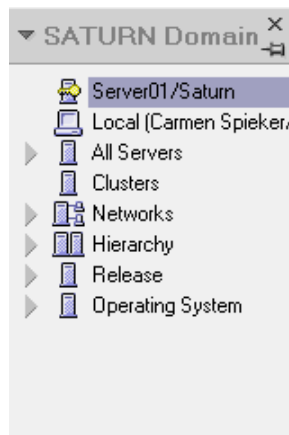


Figure 6-1 Server list for the specified domain

The bookmark bar shows two bookmark buttons that allow you to access the server list. You can add the servers you administer most often to the first bookmark button, which is for your “Favorites.”

The second bookmark is the Domain button, which displays all servers in the specified domain.

Click one of the bookmark buttons to display the server list. All servers are grouped by category, which makes it easy for you to select the server you want to administer.

Tip: Choose **Administration -> Refresh Server List** to update the server list.

6.1.4 Setting administration preferences

Setting administration preferences for the Domino Administrator makes it easier for you to administer your Domino servers. Administration preferences are in addition to the standard user preferences.

Accessing Administration Preferences

Choose **File -> Preferences -> Administration Preferences**. The Administration Preferences dialog box appears, similar to the one shown in Figure 6-2 on page 140.

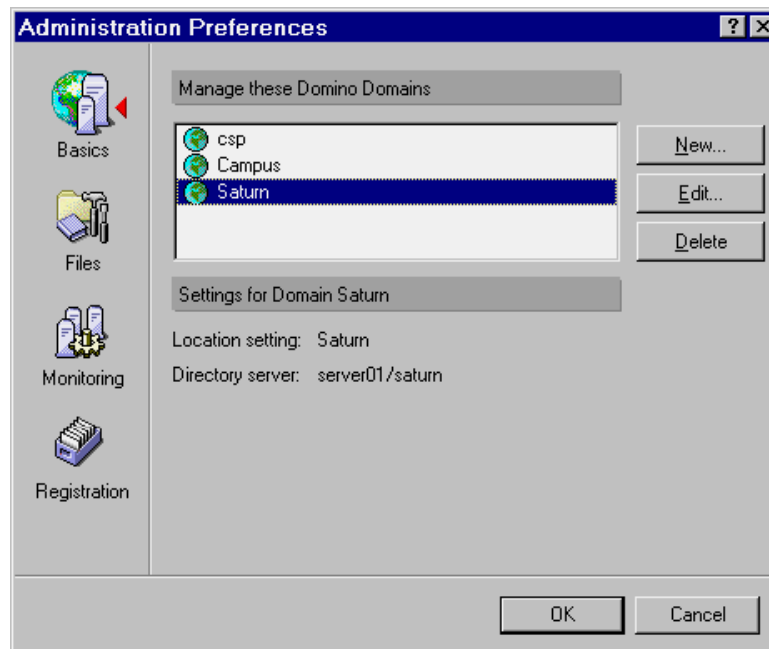


Figure 6-2 Administration Preferences

The Administration Preferences dialog box is separated into the following four pages:

- ▶ Basics

The Basics settings allow you to define which Domino domains you want to manage from the Domino Administrator client. You can choose to switch locations when you switch domains.

- ▶ Files

The Files settings let you select the columns to be displayed in the Files tab of the Domino Administrator client.

► Monitoring

The Monitoring settings allow you to configure the collection of monitoring data from the servers you administer, including how often and from which server to collect the data.

► Registration

The Registration settings allow you to set the defaults for the registration process. You can specify a default certification server and certification ID, several mail options, and the locations for ID files for users and servers.

Adding domains to your Bookmark bar

1. Choose **File -> Preferences -> Administration Preferences**.
2. Click Basics and then click New to add a domain. The Add Domain dialog box is displayed, as shown in Figure 6-3.

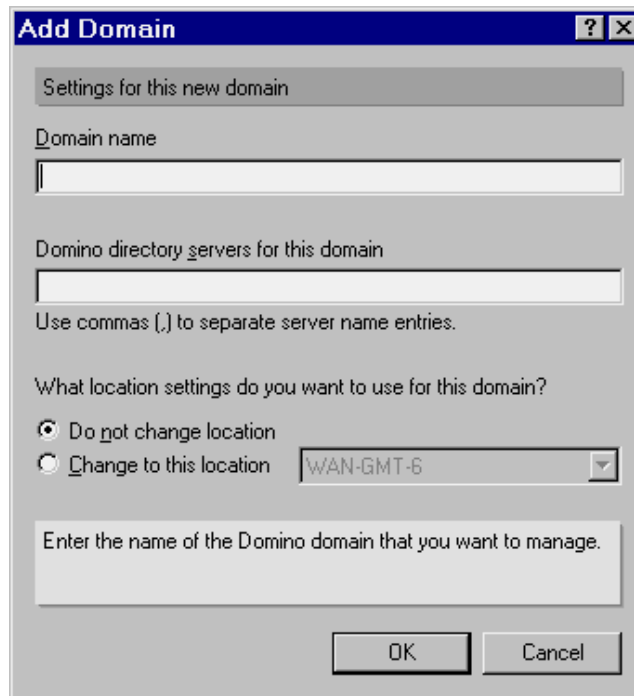
The image shows a Windows-style dialog box titled "Add Domain". It has a blue title bar with a question mark icon and a close button. The dialog is divided into several sections. The first section is labeled "Settings for this new domain". Below this, there is a label "Domain name" followed by a text input field. The next section is labeled "Domino directory servers for this domain" followed by another text input field. Below this, there is a note: "Use commas (,) to separate server name entries." The following section is labeled "What location settings do you want to use for this domain?". It contains two radio buttons: "Do not change location" (which is selected) and "Change to this location". To the right of the second radio button is a dropdown menu showing "WAN-GMT-6". At the bottom of the dialog, there is a text input field with the placeholder text "Enter the name of the Domino domain that you want to manage." and two buttons: "OK" and "Cancel".

Figure 6-3 How to add a new Domain

In the Domain name field, enter the name of the domain.

3. In the Domino directory servers for this domain field, enter one or more directory servers, separated by commas.

4. If you want to switch to a location automatically when you choose the domain, select “Change to this location,” then select the location.
5. Click OK.

6.1.5 Using tabbed pages

The middle pane in the Domino Administrator client contains tabbed pages. Each tab represents a general administration task you are likely to perform. Click a tab to display its contents or use the Administration menu to navigate among the tabs. For example, to move from the Files tab to the Replication tab, choose **Administration -> Replication**.

Most tabs have tools that appear on the right side of the Domino Administrator. The available tools change based on which tab you click. For example, if you click the Files tab, the following tools appear: Disk Information, Directory, and Database.

Tip: You can also right-click on certain objects to access the pertinent tools. For example, right-click a Person document to access People tools.

People & Groups tab

The People & Groups tab is displayed by default when you open the Domino Administrator, as shown in Figure 6-4 on page 143. You can use the tools associated with this tab to register new users, rename or recertify users, move or delete users, and set or validate the Internet address.

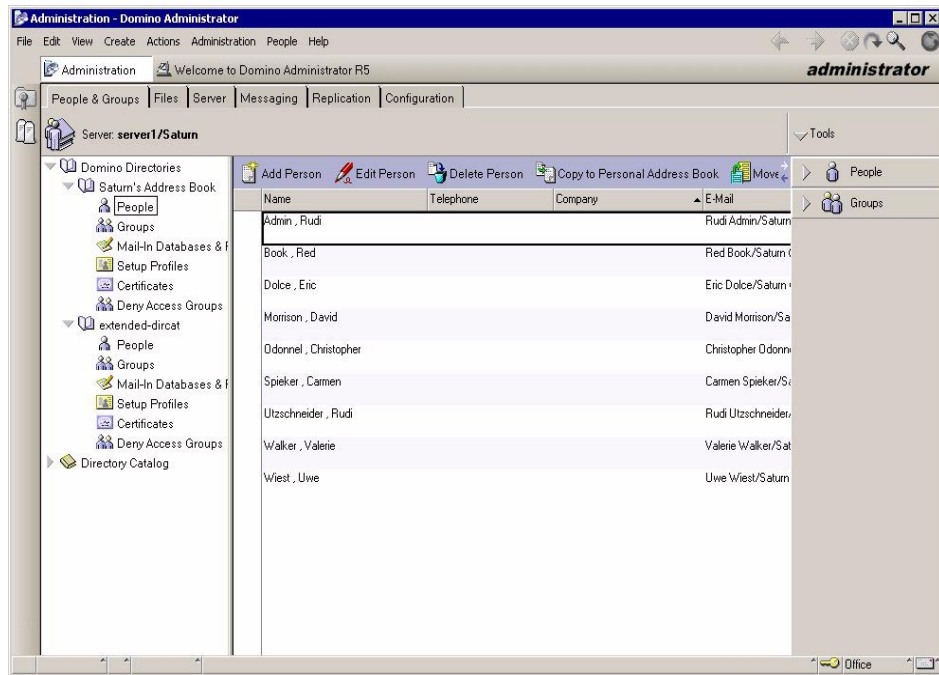


Figure 6-4 Domino Administrator People & Groups tab

In addition, you can access the tools for managing groups, as shown in Figure 6-5 on page 144. From this screen you can:

- ▶ View the membership of multiple groups while adding and removing users.
- ▶ Sort the groups by their purpose.
- ▶ Select a user and display only the groups to which the selected user belongs.

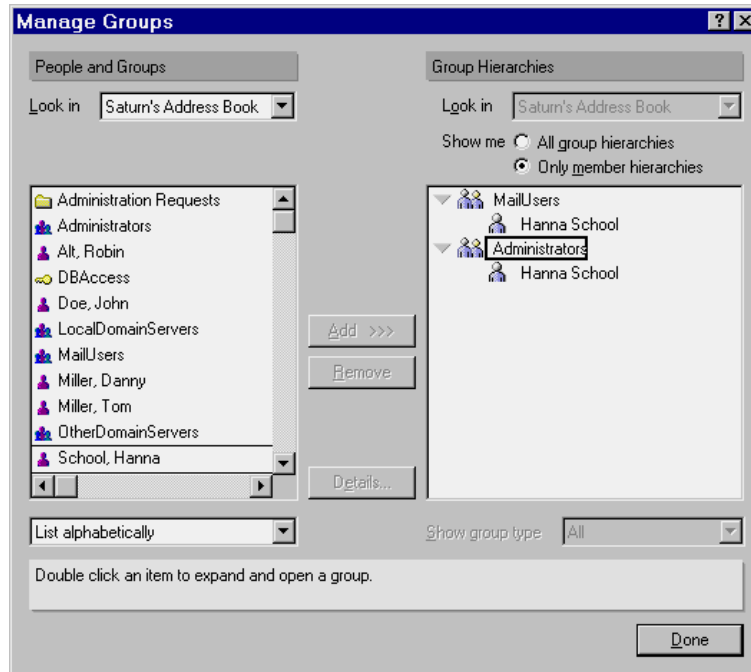


Figure 6-5 Managing Groups

Files tab

You can use the Files tab to get access to:

- Databases
- Templates
- Flat files

The Files tab provides you with a simple way to get an overall picture of your data directory structure, including database and directory links. You can check the hard disk space to see how much free disk space is available. This information is shown on the right side of the screen in Figure 6-6 on page 145.

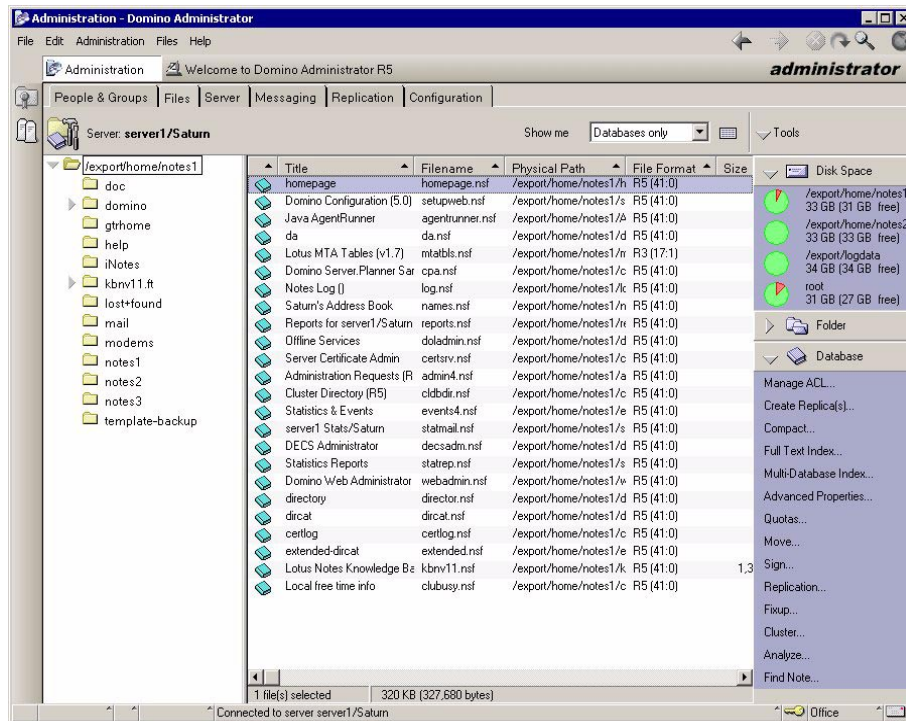


Figure 6-6 Domino Administrator Files tab

Use the Folder tools to create new folders or directory links or to delete folders.

The Database tools assist you in performing day-to-day administrative tasks, such as:

- ▶ Managing ACLs
- ▶ Creating replicas
- ▶ Compacting, moving, fixing, analyzing, or signing databases
- ▶ Managing full-text indexes
- ▶ Setting advanced properties or quotas
- ▶ Finding a specified document in a database
- ▶ Enabling or disabling databases for cluster replication

Note: You can view the ACLs of multiple selected databases. Figure 6-7 on page 146 shows the Multi ACL Management window.

You can modify the ACL of multiple databases simultaneously by using the Manage ACL tool. Be aware that you can only add, remove, or rename users in the ACLs this way.

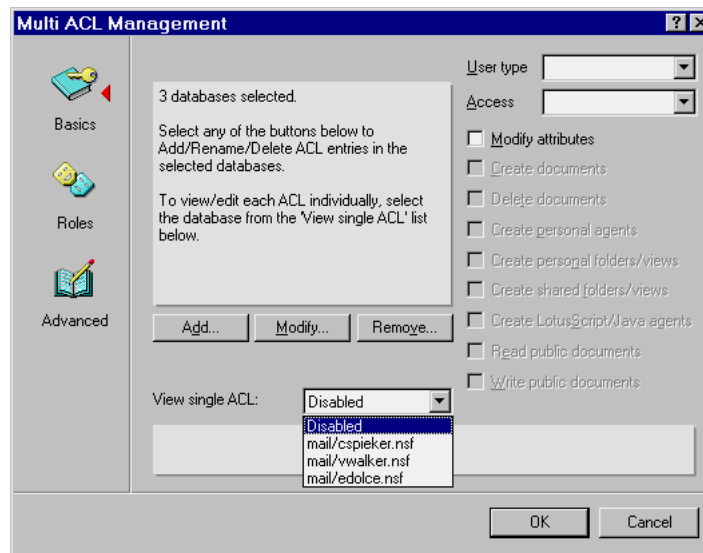


Figure 6-7 Managing ACLs of multiple databases

Tip: You can copy the ACL of a selected database by using the right-click context-sensitive menu.

Use the Compact tool not only for the compaction of selected databases, but also to kick off the archiving process. Before using this tool for archiving, you must configure the database for archiving. Use the Archive Settings button in the Database Properties InfoBox to configure archiving.

To open a Database or Template just double-click it; it opens directly from the Domino Administrator.

Server tab

Use the Server tab to administer current server activity and tasks. The Server tab is additionally broken down into four sub-tabs:

- ▶ Status
- ▶ Analysis
- ▶ Monitoring
- ▶ Statistics

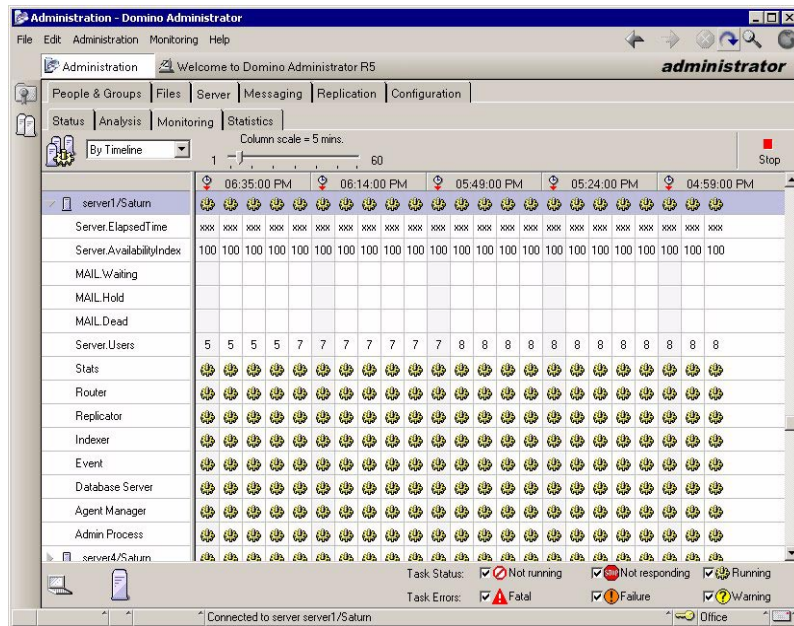


Figure 6-8 Domino Administrator Server tab

The Status tab gives you an overview of the current task and user activity. You can run a remote server console by clicking the Console button. This allows you to both start and stop server tasks, and to issue **Tell** commands.

From the Analysis tab you can analyze your server log files or cluster and start a tool that assists you in planning the decommissioning of a server.

You can easily monitor server tasks and real-time system statistics by using the Monitor tab. Status indicators show the current status of each task that you monitor.

The Statistics tab shows a list of available statistics and their most current values.

Messaging tab

You can get mail-related information from the Messaging tab, as shown in Figure 6-7 on page 146. The Messaging tab is additionally broken down into two sub-tabs:

- Mail
- Tracking Center

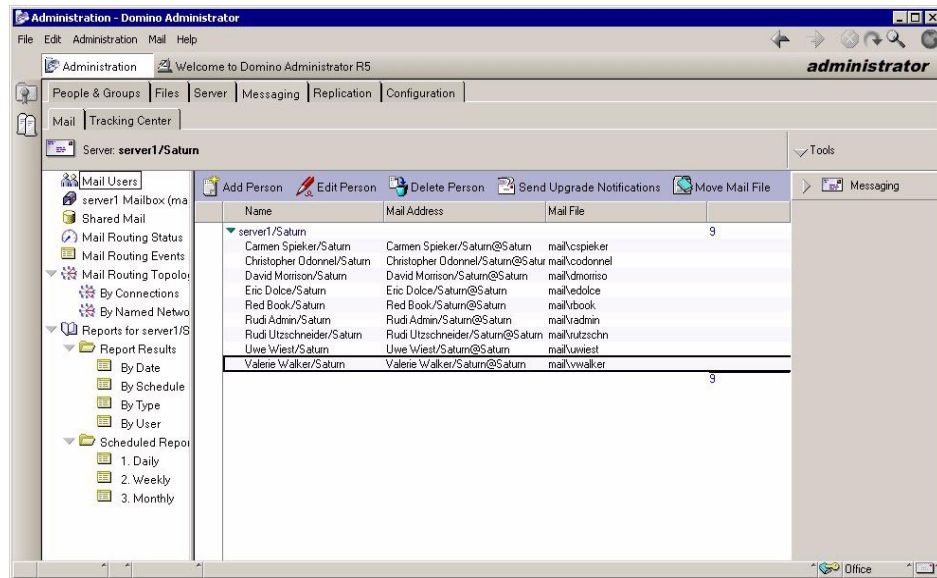


Figure 6-9 Domino Administrator Messaging tab

The Mail sub-tab allows you to view and maintain the Domino messaging infrastructure. You can use the mailbox view to delete and release dead or stopped messages, view the Mail Routing Status, or use the Mail Routing Topology view to get a graphical overview of your Mail Routing topology. To see the graphical Mail Routing Topology views, you must be running the Maps Extractor (MAPS) task on the server.

Note: You do not need access to the user's mail file to move it using "Move Mail File."

The Tracking Center sub-tab allows you to track the status of individual mail messages. You can determine if the intended recipient received the message. The tracking service must be enabled (on each server) first.

Replication tab

The Replication tab provides you with replication-related information. It is the only tab that doesn't have any specific tools associated with it. As shown in Figure 6-10 on page 149, you can view:

- ▶ A graphical representation of the current server's replication schedule.
- ▶ The Replication events view from the log file.
- ▶ A graphical representation of your Replication topology.

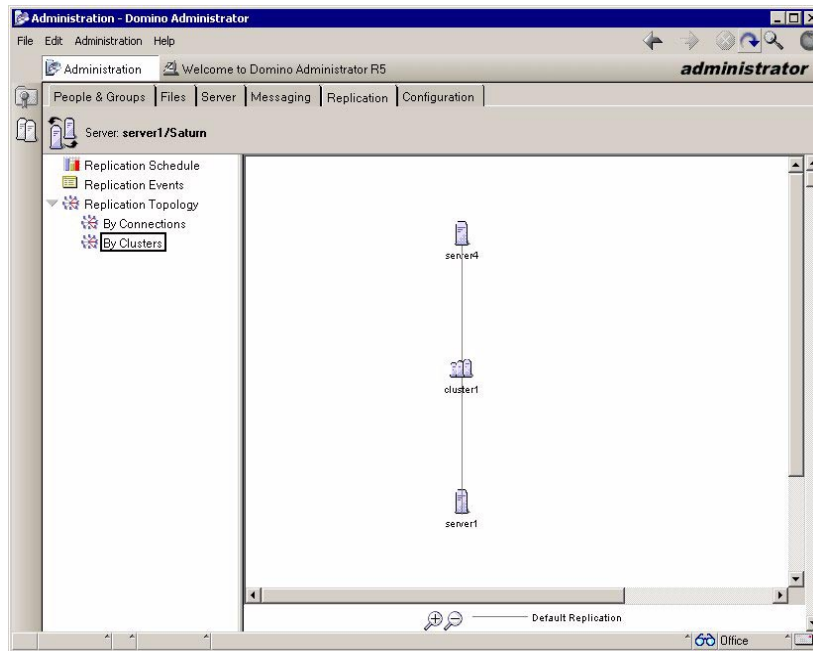


Figure 6-10 Domino Administrator Replication tab

Configuration tab

The Configuration tab, shown in Figure 6-11 on page 150, gives you access to all server configuration documents. You can access Server, Connection, Configuration, Program, and External Domain Documents under the Server category. You can also find documents related to Messaging, Replication, Web, Clustering, Statistics and Events, and the Directory itself under the appropriate categories.

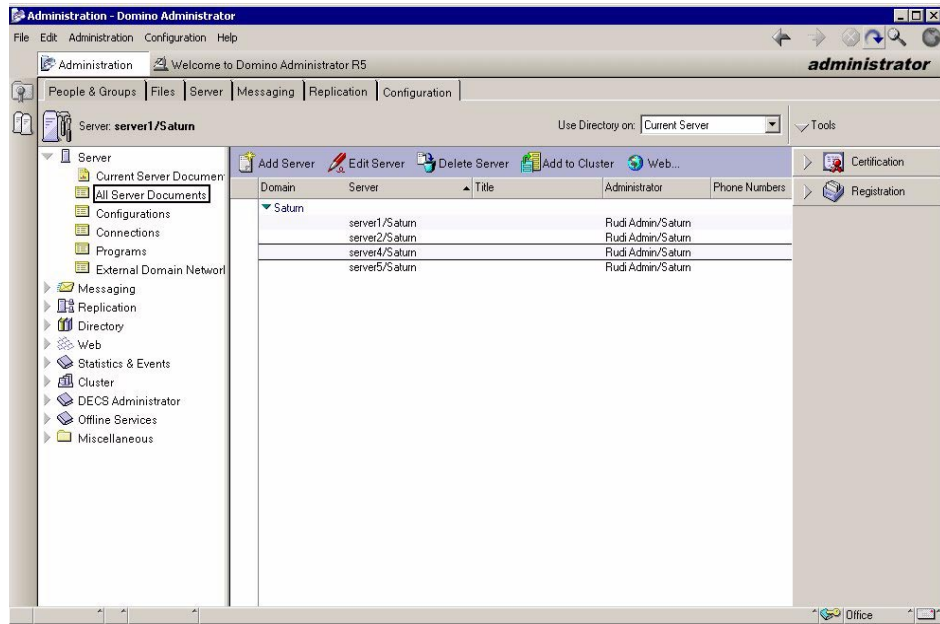


Figure 6-11 Domino Administrator Configuration tab

Use the Certification tools to certify, cross-certify, edit multiple passwords, and edit and extract recovery information. In addition, you can open the Certification Log file and view the properties of an ID file.

With the Registration tools you can register people, servers, organizations, organizational units, and Internet certifiers.

6.2 The Web Administrator

The Web Administrator allows you to manage and view settings for a Domino server from a browser. It helps you to keep your Domino server up and running, even if you don't have access to a Domino Administrator client.

In this section we present an overview of the things you can and cannot do with the Web Administrator, as well as how you can work with it.

6.2.1 Web Administrator functionality

You can use the Web Administrator to:

- Monitor mail, server memory, server disk space, and Web server statistics

- ▶ View server information and add and edit Server documents
- ▶ Manage database access control lists (ACLs)
- ▶ Create and delete databases
- ▶ Compact databases
- ▶ Create, update, and delete full-text indexes for databases
- ▶ Replicate databases
- ▶ Create new copies of databases
- ▶ Create new replicas of databases
- ▶ View database sizes and usage
- ▶ Manage users and groups
- ▶ View server log information and monitor messages in the log
- ▶ View database catalog information
- ▶ View events and statistics reports in the Statistics database
- ▶ Use the remote server console
- ▶ View Administration Process requests in the Administration Requests database
- ▶ Track messages
- ▶ Edit text files, for example Notes.ini

6.2.2 Web Administrator limitations

Keep the following limitations in mind before using the Web Administrator:

- ▶ The Server you administer must be set up as a Web server, running the HTTP task.
- ▶ You can only administer the server on which the Web Administrator runs.
- ▶ You cannot use passthru to access another server using the Web Administrator.
- ▶ You cannot register users, servers, or certifiers.

6.2.3 Working with the Web Administrator

In this section we describe how to work with the Web Administrator.

Prerequisites

Before you can start setting up and using the Web Administrator, the following prerequisites have to be met.

The HTTP task must be running

Before you can start working with the Web Administrator, you have to make sure that the server you want to administer is set up as a Domino Web server. Although you can use the server for other server tasks—for example, mail routing and directory services—you must run the HTTP task to use a browser to access the server. For information on setting up a Domino Web server, see Chapter 9, “Domino R5 as a Web server” on page 227.

Several specific databases must exist

The Web Administrator accesses system databases to provide its functionality and to view information. You can only use all Web Administrator functions if the following databases exist on your server:

- ▶ Administration Requests (Admin4.nsf)
- ▶ Database Catalog (Catalog5.nsf)
- ▶ Domino Directory (Names.nsf)
- ▶ Log (Log.nsf)
- ▶ Domino Administration Help (Help5_Admin.nsf) in the Help subdirectory on the server
- ▶ Statistics (Statrep5.nsf)
- ▶ Statistics & Events (Events5.nsf)
- ▶ Web server log (Domlog.nsf)

You must have an Internet password

You need your user name and Internet password to authenticate with the server. Set the Internet password in your Person document in the Domino Directory. You can find it on the Other tab.

Setting up the Web Administrator

With the Web Administrator, all you need to administer your server is a network connection to the server and a browser. You will use your network connection and browser to access a database in the Domino data directory called Web Administrator (Webadmin.nsf). This database is created automatically when you first start the HTTP task.

Note: Each Web Administrator database has a unique replica ID. Therefore, it cannot replicate between servers.

In order to run the Web Administrator, you need to have access to the Webadmin.nsf database.

Setting up access to the Web Administrator database

Domino automatically gives access to all names listed in the Administrators field of the Server document when Domino creates the Web Administrator database. To allow additional administrators to use the Web Administrator, you must give them the appropriate access.

Perform the following steps to set up access for another server administrator (individual person) or group of server administrators:

1. Add the server administrator's name or the group name to the ACL of Webadmin.nsf and give the user or group Manager access.
2. Refine access by assigning the appropriate Web Administrator database roles. Select the ServerAdmin, ServerMonitor, DatabaseAdmin, FileRead and/or FileModify roles.
3. Edit the Server document for the Domino Web server in the Domino Directory and add the user or group name to the "Administer the server from a browser" field on the Security tab.
4. Add the user or group name to the "Run restricted LotusScript agents" field in the Agent Manager section of the Server document.
5. Make sure that all server administrators have an Internet password set in their Person document.

Note: If you're using Secure Sockets Layer (SSL) for authentication, you have to set up the browser for SSL. For more information, see "Setting up clients for SSL client authentication" in the Domino 5 Administration on-line help database.

Controlling access to the Web Administrator files

The Web Administrator files are stored in the subdirectory domino/adm-bin of the Domino data directory. You should protect access to this directory so that unauthorized users cannot access the Web Administrator. Domino creates a File Protection document for this subdirectory by default when you start the server for the first time. The File Protection document appears in the Web - Web Server Configurations view on the Configuration tab of the Domino Administrator. The Notes.ini file variable DominoConfigLevel indicates whether the File Protection document was created during server startup.

The meanings of the DominoConfigLevel values are as follows:

- 1 Mapping/Redirection and virtual server documents in the Domino Configuration database (Domcfg.nsf) are upgraded, but the File Protection document for domino/adm-bin is not created yet.
- 2 File Protection document for domino/adm-bin is created, but the Mapping/Redirection and virtual server documents in the Domino Configuration database are not upgraded yet.

- 3 Both the File Protection document and upgrade of the Domino Configuration database are done.

Use the DominoConfigLevel setting to troubleshoot problems with controlling access to the Web Administrator files and migrating Mapping/Redirection and virtual server documents.

Do not modify this setting in the Notes.ini file.

Accessing the Web Administrator

Use the following steps to access the Web Administrator:

1. Open the Web Administrator by entering the following URL into your browser:

`http://hostname/webadmin.nsf`

where *hostname* is the hostname or IP address of the Domino Web server you want to administer (for example,

`http://myserver.company.com/webadmin.nsf`)

2. Enter your *user name* and *your Internet password*, where *user name* is your hierarchical name or common name (for example, John Doe/Company or John Doe)

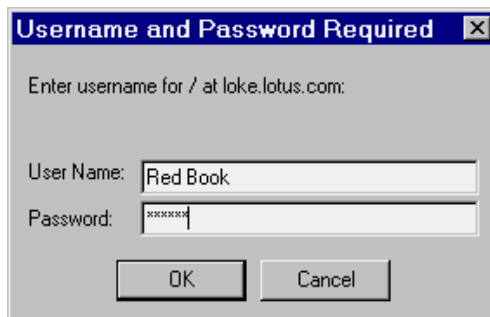


Figure 6-12 Web authentication

After you've successfully authenticated, you will see a screen similar to Figure 6-13 on page 155.

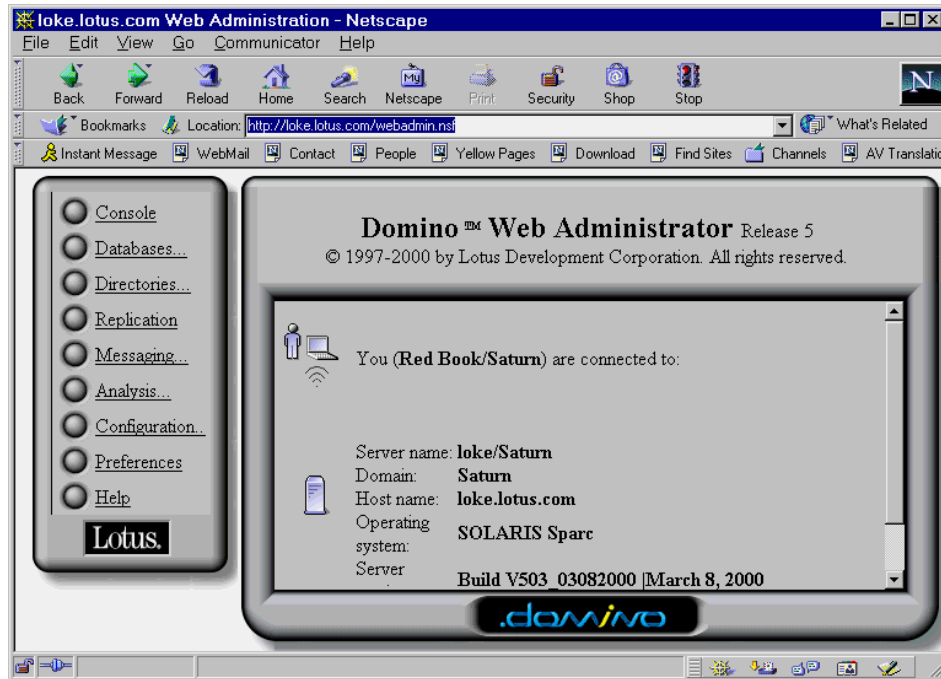


Figure 6-13 Domino Web Administrator

Tip: You can use multiple instances of the Web Administrator at the same time by starting another copy of the browser. You can arrange the windows so both copies are displayed on the screen.

Main administrator tasks

The Web Administrator offers a variety of tools which help you to administer your Domino servers. When you click on a Web Administrator task, such as Databases or Messaging, a list of more specific tasks appears. Most of the Web Administrator tools should look familiar if you are familiar with Domino Administrator. This section describes the main Web Administrator tasks.

Console

The Web Administrator allows you to open a remote console to the server, as shown in Figure 6-14. The remote console appears essentially the same as it would from the Domino Administrator client. You can select a server command or enter the console command directly to the server. You can enable the Live Console check box to see Live Console reports and commands from the server.

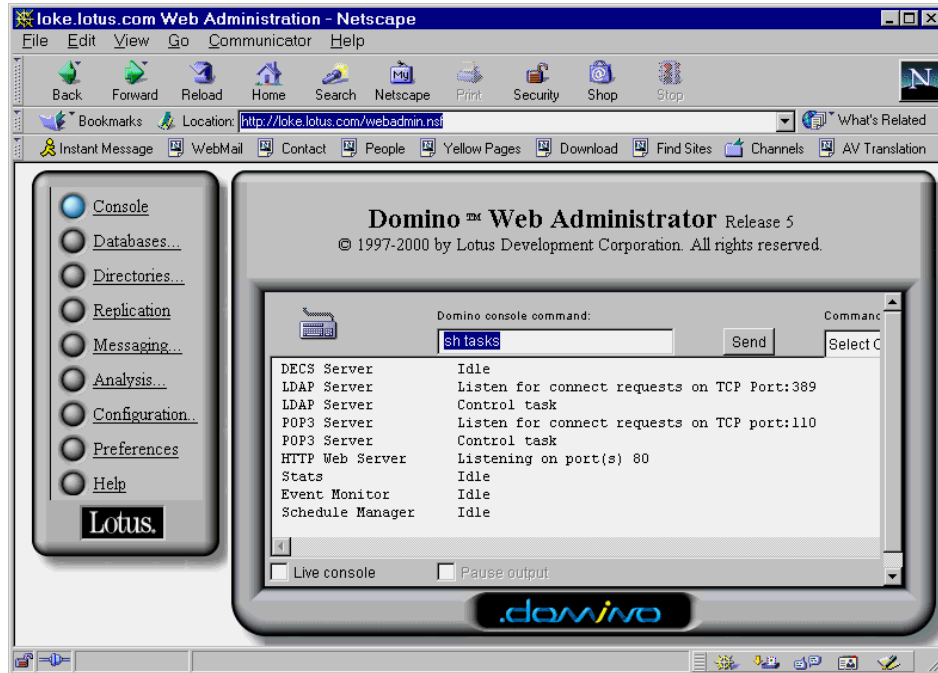


Figure 6-14 Domino Web Administrator Console

Databases

The Web Administrator lets you delete, compact, full-text index, replicate, and create a new copy or replica of a database. You can also modify a database's Access Control List and view Sizes, Usage, Catalog, or Cluster Directory. The Web Administrator also displays the following about the database:

- ▶ Full-text index information
- ▶ Current white space percentage to help you determine whether or not to compact the database
- ▶ Current database size quota

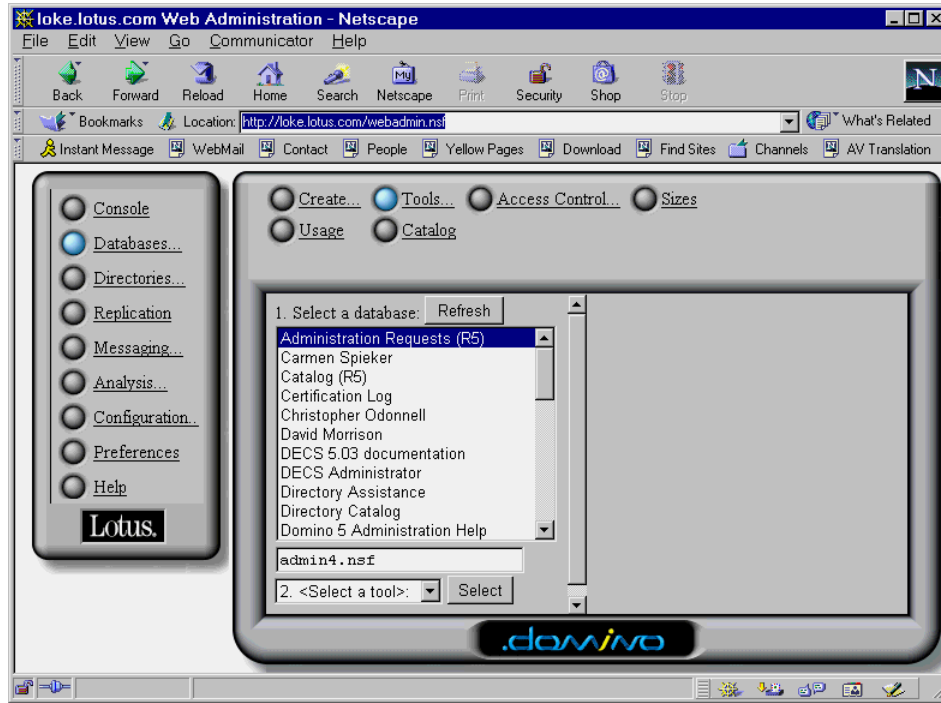


Figure 6-15 Domino Web Administrator Databases tab

Table 6-1 shows the minimum access level required for a user to access various database functions.

Table 6-1 Minimum Access Control List levels

Database action	Minimum access level
Compact	Reader
Delete	Manager
Create full-text index	Designer
Update an index	Reader
Replicate, create a new copy or replica	Reader
Make changes to ACL	Manager

In addition, the “Maximum Internet name & password access” must be set to “Manager” in the database whose ACL you are changing if you are using name-and-password authentication to access the database.

Directories

You can use the Web Administrator to create Group and Person documents in the Domino Directory. You can also view information in existing Group and Person documents, as shown in Figure 6-16.

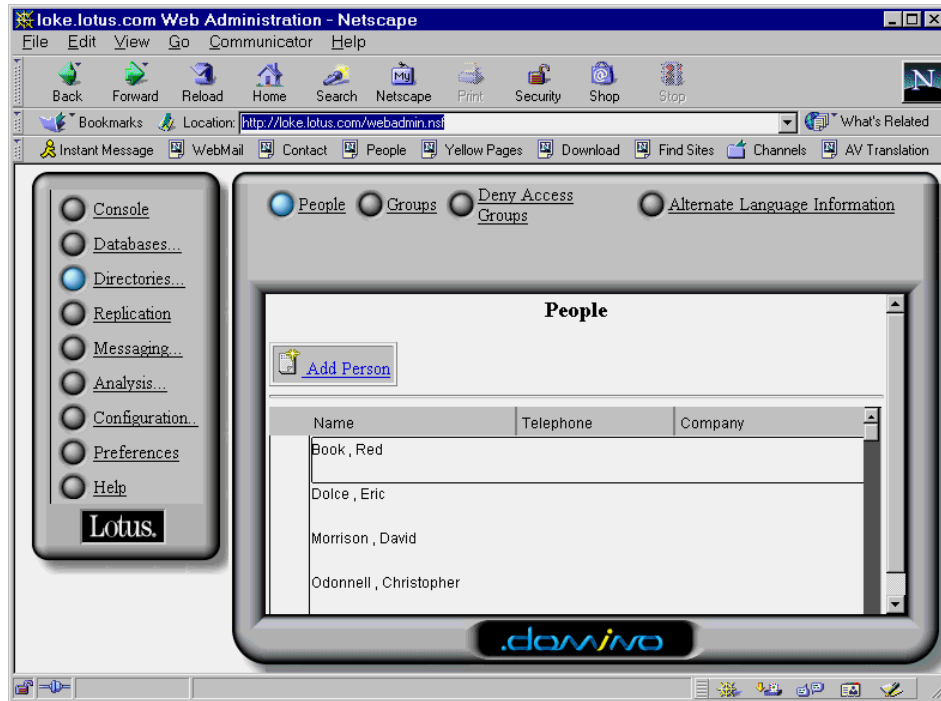


Figure 6-16 Domino Web Administrator Directories tab

Note: You can't register a user from the Web Administrator, but you can create a Person document that would allow a user to connect to the server via a browser.

Note: You must have at least Editor access, or Author access and the UserCreator role in the Domino Directory, to create new Person documents. You must have at least Editor access, or Author access and the GroupCreator role in the Domino Directory, to create new Group documents.

Replication

The Web Administrator lets you access the Replication Events view from the servers log file (Log.nsf).

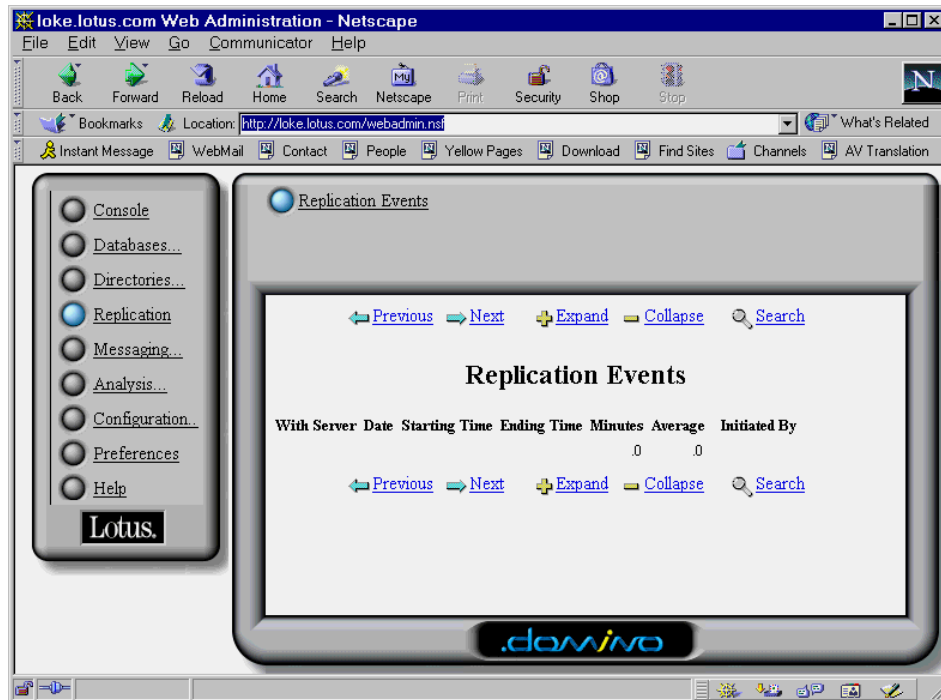


Figure 6-17 Domino Web Administrator Replication tab

Messaging

The Web Administrator graphically displays the Routing Status. You can see the number of dead and pending mail messages on the Domino server. Dead mail is the number of mail messages on the server that cannot be delivered due to a problem or error. Typically, the server should show no undeliverable messages. Pending mail is the number of mail items that the server is waiting to deliver. Typically, the server has only a few messages awaiting delivery, even during times of peak mail use.

You can access the Routing Events, Shared Mail, and Mail User views. If you have mail tracking enabled on a server, you can generate usage reports on that data. You can also track specific mail messages to determine if the intended recipient received them. Figure 6-18 shows the routing status of messages on the Domino Server. In this case there is no dead or waiting mail.

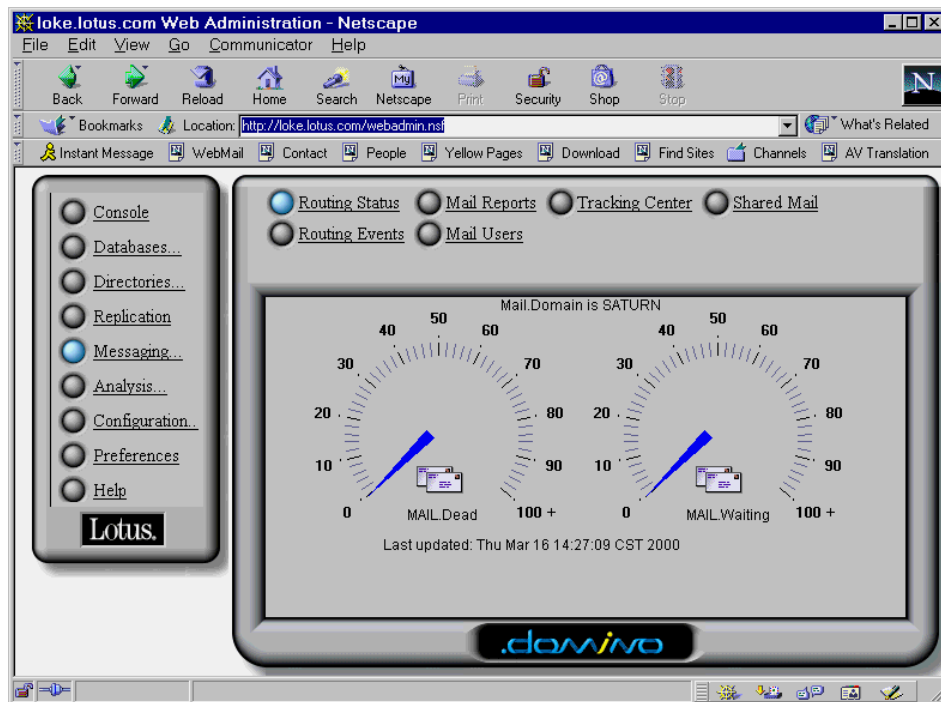


Figure 6-18 Domino Web Administrator Messaging tab

Analysis

The Web Administrator lets you access the Logfile, Memory, Statistics, Diskspace, Administration Requests, and Statistics and Events. You can also view graphics on Alerts and Web Statistics, as shown in Figure 6-19.

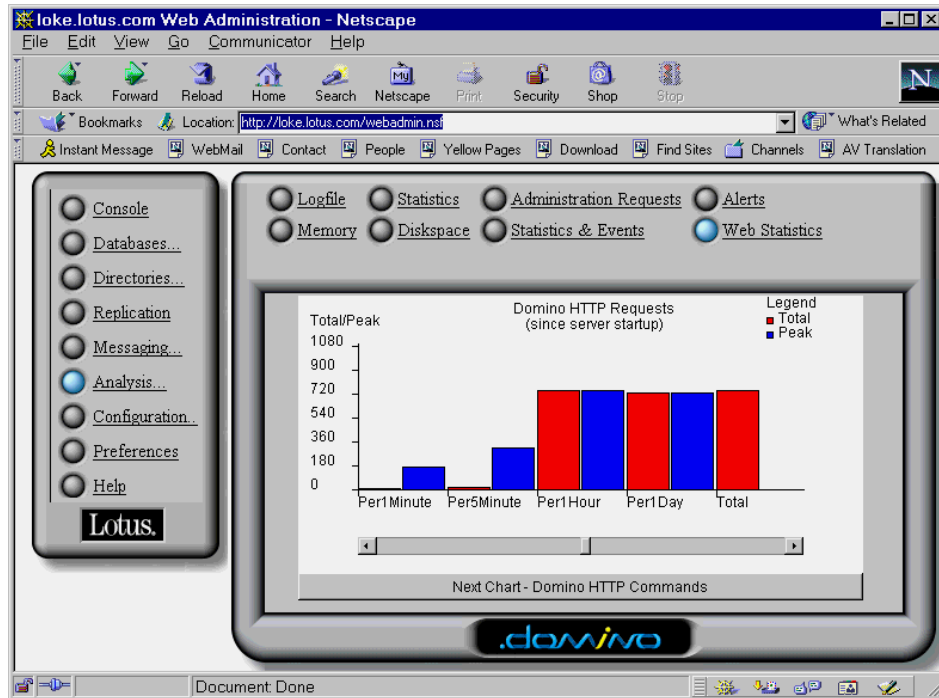


Figure 6-19 Domino Web Administrator Analysis tab

Configuration

You can access Servers (server documents), Web Configuration, Clusters, Connections, Domains, Programs and Configurations views from the Domino Directory. Figure 6-20 shows how you can open and edit System Files on the remote host system.

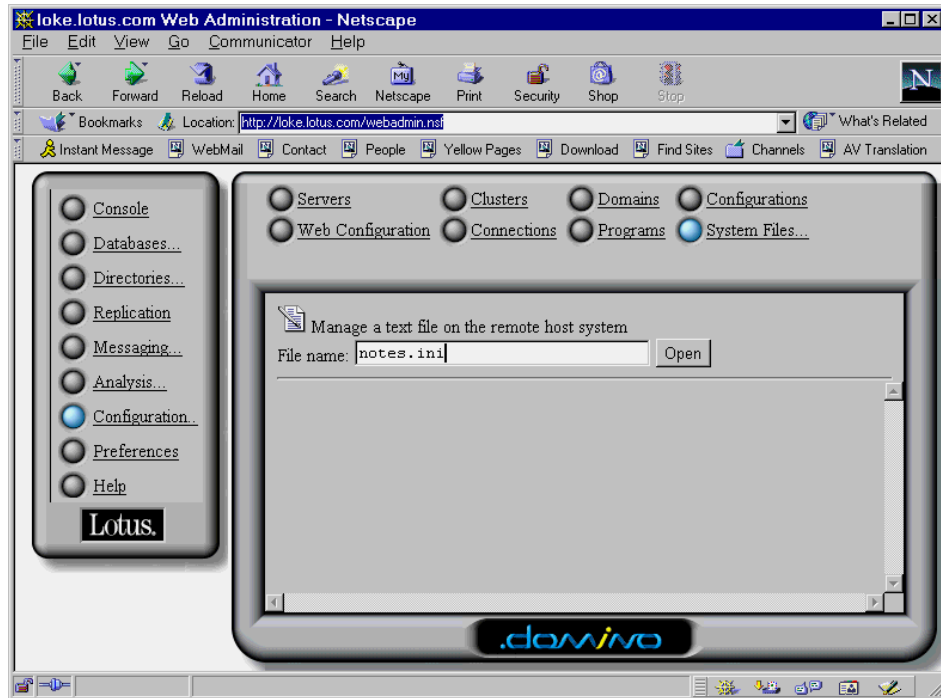


Figure 6-20 Domino Web Administrator Configuration tab

Preferences

Using the Preferences command, you can select whether you want the user interface to appear as buttons or a drop-down list. The “plain” interface displays commands in a drop-down list, and doesn’t use additional frames or graphics. Click Preferences in the navigator pane to see what your current settings are, as shown in Figure 6-21. You can edit your Preferences by clicking the Edit Preferences button/link. View Environment Information by clicking on the assigned twisty to expand this field.

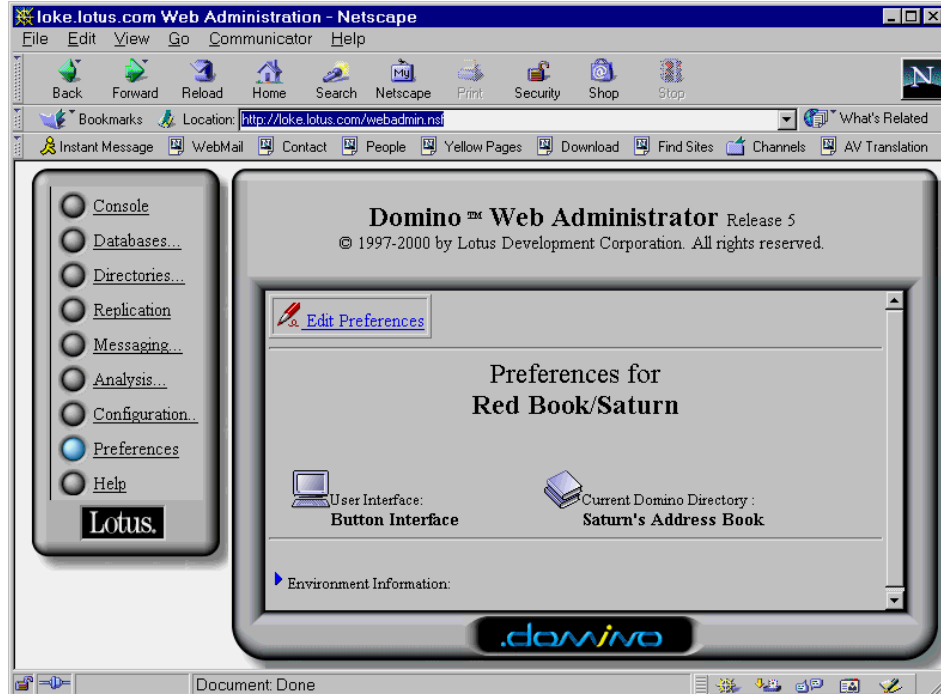


Figure 6-21 Domino Web Administrator Preferences tab

Help

You can open the Domino 5 Administration Help database at any time, from anywhere in the Administration client. To access help, click the Help button in the main navigator pane or press F1. The Domino 5 Administration Help database contains detailed information on all aspects of setting up and maintaining a Domino server.

Troubleshooting

This section describes some known problems and possible solutions. You can use this information as a basis for troubleshooting Web Administrator problems.

- ▶ If you are having trouble authenticating with the Domino Web Administrator, ensure that you are logging in with one of the entries specified in the “User name” field of your Person document, and that you are using the correct password as specified in the “Internet password” field.

Note: The short name is only valid if it is added to the “User name” field.

- ▶ Your name should also be listed “Administer the server from a browser” field in the Server document, and in the ACL of Webadmin.nsf.
- ▶ Make sure that the File Protection document for the Web Administration executables is configured properly. File Protection documents protect files on a server’s hard drive. They control the access that Web browser clients have to the files. Both the server and administrator should have post and get (read and write) access to the domino/adm-bin directory. For more information on File Protection documents, see “Protecting server files from Web client access” in the Domino 5 Administration Help database.
- ▶ Some browser configurations may require two authentications due to the way realms are handled. A realm is a string, typically a URL path, that the server sends to indicate the location, or path, for which the user has been authenticated. When two authentications are required, the remote console and some of the other applets do not function correctly until the second authentication has occurred. Selecting “Live Console” forces a second authentication if it is required.
- ▶ The Web Administrator incorporates a new feature in R5 that allows administrators to edit ASCII text files on Domino servers (such as the Notes.ini file). To enable this feature, open Webadmin.nsf with a Notes client and click the button to “Sign unrestricted agents for browser access.” You need Manager access in the ACL to do this. You also need to add the hierarchical name of the server administrator (person) to the “Run unrestricted LotusScript/Java agents” field in the Domino Server record.
- ▶ If your Domino server ID is using a password and you want to use the Web Administrator to modify database security (such as database ACLs and roles), you will need to set the server ID to “share password with Notes add-ins.” Failure to do so may cause the Domino server to hang if the Web Administrator is used to modify database ACLs. To set this, use a Notes client to access the Server ID, select File -> Tools -> UserID -> Basics, and enable the “Share password with Notes add-ins” check box.
- ▶ Web administrators cannot modify Notes database security by default. This is because Notes databases don’t generally have the “Maximum Internet name & password access” set above “Editor.” If you want to modify Notes database security from a browser, you will need to use a Notes client to change this parameter to “Manager.” You will also need to ensure that this parameter is set to “Editor” or above if you want to edit Notes documents from a browser.

- ▶ Your Web browser cache should be set to check documents every time.
- ▶ Internet Explorer 4.x users need to disable HTTP 1.1. This setting is located at the bottom of the Advanced section under View -> Internet Options -> Advanced.
- ▶ If your Internet browser setting in your Location document is set to Notes with Internet Explorer or Microsoft Internet Explorer, you must make sure the Update cache setting in the Advanced section of the Location document is set to “every time.” This setting overrides the setting in Microsoft Internet Explorer.
- ▶ Domino Web Administrator 5.0 database tools don’t recognize group membership in database ACLs.
- ▶ If you have trouble creating databases, or replicas of databases, using the Web Administrator, it may be because the “Create new databases” or “Create replica databases” field in the Server document is not set correctly.

Tip: If you encounter any problems with the Web Administration application, it is possible that the database has become damaged. Shut down HTTP, delete Webadmin.nsf, and restart HTTP. When HTTP starts it checks for the existence of Webadmin.nsf, and creates a new one from the template if it doesn’t exist.

6.3 The Domino Character Console

The Domino Character Console is a new feature in Domino R5. It is only available on UNIX platforms, although there’s something very similar for Domino on the AS/400 which has been there since 4.6.

The Domino Character Console (the cconsole program) provides a way to access the server console from the UNIX command line. You can invoke the cconsole program multiple times simultaneously. You can also run the cconsole program when there is already an operational Domino server console; however, the cconsole input and output may also reflect commands launched from other console processes.

Note: The cconsole program is installed into your Domino bin directory, /opt/lotus/bin

6.3.1 Starting Domino Character Console

The Domino Character Console is started by changing the current directory to the Domino data directory and running `/opt/lotus/bin/cconsole`. Table 6-2 shows the command line switches.

Table 6-2 Domino Character Console switches

Switch	Result
-f <i>ID_file</i>	Specify the ID file to use.
-i	Ignore warnings .
-l	Start cconsole with Live Console activated.

For example, to start the cconsole and use admin.id as the user ID file so that there is no prompt, you would type:

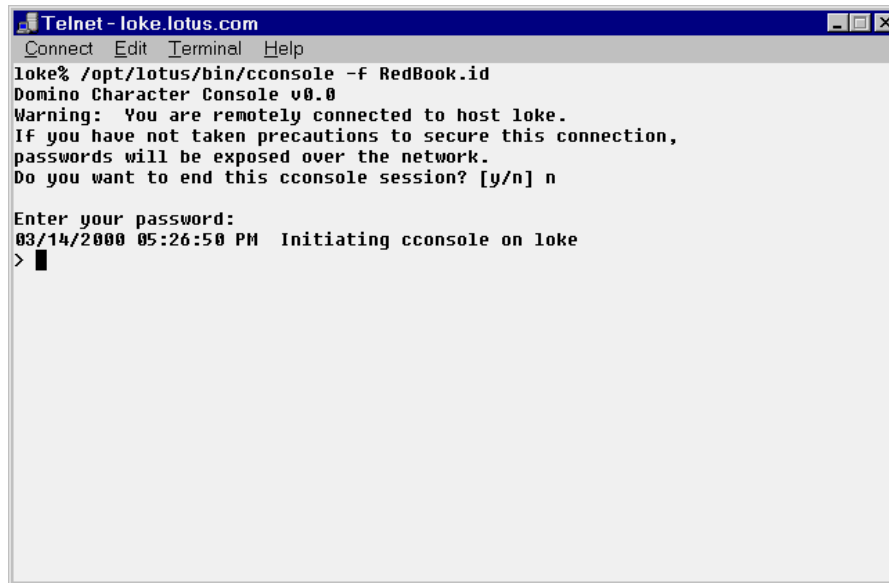
```
/opt/lotus/bin/cconsole -f /server1/admin.id
```

and then press Enter. Make sure that the ID file has appropriate read and write file permissions.

By default, there is no admin.id in your Domino data directory. You have to use FTP to transfer your ID file from your local drive to the Domino data directory. Use binary mode for the transfer.

Note: There is a security risk when running the cconsole program from a remote machine or from a remote X Windows display. The cconsole program warns you of this security risk before proceeding. Deploy a secure remote protocol, such as encrypted telnet. To address this security risk, if you don't deploy a secure remote protocol, run the cconsole program only from the local Domino server.

Figure 6-22 on page 167 shows the cconsole after successful login with a user ID.



```
Telnet - loke.lotus.com
Connect Edit Terminal Help
loke% /opt/lotus/bin/cconsole -f RedBook.id
Domino Character Console v0.0
Warning: You are remotely connected to host loke.
If you have not taken precautions to secure this connection,
passwords will be exposed over the network.
Do you want to end this cconsole session? [y/n] n
Enter your password:
03/14/2000 05:26:50 PM Initiating cconsole on loke
> █
```

Figure 6-22 Running cconsole

From the > prompt you can now enter console commands, such as *show tasks*.

6.3.2 Stopping Domino Character Console

To stop the cconsole:

1. Type **done**
2. Press Enter.

Note: Do not type **quit**, **q**, **exit**, or **e**. These are the commands to shut down the Domino server.

Figure 6-23 on page 168 shows the display after stopping the cconsole.

```

Telnet - loke.lotus.com
Connect Edit Terminal Help
LDAP Server          Control task
POP3 Server          Listen for connect requests on TCP port:110
POP3 Server          Control task
HTTP Web Server      Listening on port(s) 80
Stats               Idle
Event Monitor        Idle
Schedule Manager     Idle
Calendar Connector   Idle
Admin Process        Idle
Agent Manager        Executive '1': Idle
Agent Manager        Idle
Indexer              Idle
Replicator           Idle
Router              Idle

> sh us
  User Name          Databases Open      Minutes Since Last Used
Carmen Spieker/Users/Saturn
                        admin4.nsf          1
                        names.nsf           4

> done
03/14/2000 05:34:17 PM Ending cconsole on loke.
loke%

```

Figure 6-23 Stopping cconsole

6.3.3 Commands

In addition to the usual Domino Server console commands, the character console supports a few additional ones, shown in Table 6-3.

Table 6-3 Additional console commands using the Domino Character Console

Command	Result
done	Exit the console and leave the server running.
live on	Enable live display of console activity.
live off	Disable live display of console activity.

6.4 Summary

In this chapter we have shown how to connect to your Domino servers using different administration tools: the Domino Administrator client with its tabbed pages, the Web Administrator for administering your server from a browser, and the Domino Character Console for entering server console commands from the command line.



Security

While the security of Solaris and Lotus Domino are both very extensive topics that cannot be completely covered in this book, this chapter offers a starting point for considering how to implement a secure Solaris system that runs Lotus Domino.

It is divided in two parts: the Solaris operating environment and the Lotus Domino environment.

The Solaris operating environment part focuses on which services you might consider disabling to make a system secure.

The Domino part starts with an overview of Domino and Notes security basics, followed by a discussion on how to protect your Domino server and databases.

7.1 Solaris operating environment security

The Solaris operating environment is a flexible, general purpose operating system. Due to its general nature, changes must be made to secure the system against unauthorized access and modification. This section discusses the Solaris operating environment subsystems and the security issues surrounding them.

Recommendations are made on the manner in which each of the subsystems should be secured. As with any security decisions, a balance must exist between system manageability and security.

Our discussion of the Solaris operating environment system security is organized into two main parts: file system and local security, and security of network services. We will also list the most common security tools and where to find them.

The information applies to the Solaris 7 and 8 operating environments. If you need more security than C2 Sun provides, a version of Solaris that was designed to meet the TCSEC CMW (Orange Book C2 Compartmented Mode Workstation) is available; it is called Trusted Solaris. For more details, see the Security section of the Sun Web site at www.sun.com.

7.2 File systems and local security

Often, administrators are mainly concerned about unauthorized access to their systems by people logging in remotely. However, there should be the same concern for local, authorized users gaining extra privileges on a system by exploiting a problem with internal system security.

7.2.1 Solaris patches

Building a secure Solaris operating environment involves installing a new system with the latest version of the Solaris operating environment and applying the latest patches. Each new release of Solaris includes security improvements and additional features to enhance system security. It is recommended that you always use the latest version of the Solaris operating environment that your applications will support.

Sun provides patches to the Solaris operating environment and Unbundled Software Products when problems are found and corrected. Anyone can download recommended security patches for the Solaris operating environment; see <http://www.sunsolve.sun.com/> and the official Sun FTP site for more details. All other patches require a SunSpectrum service contract.

All systems should have the latest recommended security patches installed. Subscribe to the Sun security bulletin mail list to receive notification of important security-related patches. See the Sun Web site for more information.

Sun provides Maintenance Updates (MU) for the Solaris operating environment. An MU is a tested combination of patches for a specific release of the Solaris operating environment that installs in one quick and easy step. These updates are only available to service contract customers. SunSpectrum service contract customers have access to all patches, maintenance updates, and the patchdiag tool.

The patchdiag tool is very useful. It takes a list of current patches available from Sun and examines the local system to determine patches that have not yet been applied. It also checks for new versions of patches that have already been applied. The patchdiag tool should be run on the system at least once a week to determine if important patches need to be applied.

To check which patches are installed in the system use the Solaris command:

```
# showrev -p
```

7.2.2 File system

The Solaris operating environment ships with some file system permissions that should be adjusted for security reasons. For example, many files and directories have the group write bit set. In most instances, this permission is not necessary and should be switched off.

File permissions

You can see the file permission using the **ls -la** command; the information is in the first column. For example, to see the permission of the file `billing`, use the command:

```
# ls -la billing
-rwxr-xr-x 1 root  bin  42564 Dec 16 12:37 billing
```

In the permission string `-rwxr-xr-x` the first position tells you whether it is a directory (d), a file (-), or a link (l). The next nine bits are divided into three groups of three (rwx). They refer to the owner (you), the groups you belong to, and finally, others (meaning everyone else). You can grant members of these three groups the ability to read (r), write (w), or execute (x) a file or directory. Refer to the `chmod` command for more details.

Tip: There is a free third party tool, created by Casper Dik, to adjust these permissions. The tool is called `fix-modes`.

File system partitions

When creating operating system file partitions, be sure to allocate adequate disk space for system directories, log files, and applications. Certain server applications or services may require extra disk space or separate partitions to operate effectively without impacting other services.

Typically, there should be separate partitions for:

- ▶ / (the root file system)
- ▶ /usr
- ▶ /var
- ▶ /opt

Note: By default, the Lotus Domino programs directory is installed under /opt.

Usually the Solaris operating environment /var file system contains:

- ▶ System log files
- ▶ Patch data
- ▶ Print, mail, and files for other services

The disk space required for these files will vary over time. Most systems should maintain /var as a separate partition from the root file system. Provide extra space in /var if you intend to store large log files.

The set-user-ID and set-group-ID bits

The set-user-ID and set-group-ID bits (sometimes referred to as SUID and SGID bits) on an executable file indicate to the system that the executable should operate with the privileges of the file's owner or group, for example:

```
-r-sr-xr-x 1 root bin 24270 Dec 16 12:38 bindsock
```

The flag `s` in the 4th position shows that the file is a set-user-ID.

An attacker can use the elevated privileges provided by the set-user-ID or set-group-ID mechanism to execute code on the program stack (a “buffer overflow” attack) or to overwrite system files. When these security problems are reported, Sun identifies a fix for them and provides a patch. This is another reason to keep your system up to date with the latest set of patches.

The Lotus Domino bindsock program uses the `suid` as root. This is done to permit Domino to open ports less than 1024 to the Internet Domino processes, such as IMAP, POP3, HTTP.

To find all the set-user-ID and set-group-ID files on a server, use the following find command:

```
# find / type f \( -perm -u+s -o -perm -g+s \) -ls
```

Note: Executing this command in the Domino program directory should give the following output (see 9.5.3, “Bindsock issue” on page 247 for more details on the bindsock program):

```
373631 15 -r-sr-xr-x 1 root bin 14504 Dec 16 18:48 ./bindsock
```

7.2.3 Mounting file systems

The Solaris operating environment file system partitions can be mounted with various options that enhance security. To prevent an attacker from using the set-user-ID feature, you can mount the file system in one of the following ways:

- ▶ Use the nosuid option
- ▶ Mount the file system in read-only mode

Basic recommendations

It is not possible to mount all the file systems with these two options. For example, the /usr partition can be mounted read-only, but it should not be mounted nosuid since there are some commands in this partition that have the set-user-ID bit set.

The same thing is valid for /opt for Domino: it needs to have the bindsock program using the suid.

The /var partition cannot be set to read-only, but can be set to nosuid. All other partitions should be mounted read-only and with nosuid whenever possible.

It is not possible to mount the root file system (/) with the nosuid option on modern releases of the Solaris operating environment. This is due to the fact that the root file system is mounted read-only when the system boots and is later remounted read-write. When the remount occurs, the nosuid option is ignored.

The vold daemon

The Solaris Volume Management system provides users with an easy way to mount removable media without requiring superuser access. CDROMs and floppy disks are mounted and unmounted automatically by the volume management system.

The daemon that manages this system is called vold. Unfortunately, the Volume Management system allows suid file systems for all removable media that are capable of supporting it. Anyone can insert a UNIX File System (UFS) formatted floppy disk containing a set-user-ID executable and gain control of the system. To prevent this situation, add the following lines to the end of the `/etc/rmmount.conf` file:

```
mount hsfs -o nosuid
mount ufs -o nosuid
```

To install Domino it is necessary to mount the CDROM, which you can do manually with the command:

```
# /etc/mount -r -F hsfs /dev/dsk/c0t6d0s2 /cdrom
```

The name of the cdrom driver, `c0t6d0s2` in this example, may be different in your system.

7.2.4 Accounts

Managing user and system accounts is an important aspect of Solaris operating environment security. A default Solaris operating environment installation contains several accounts that must be either deleted or modified to strengthen security.

Basic recommendations

Some accounts are not necessary for normal system operation. These accounts include:

- ▶ `smtp` - the mail transfer protocol user
- ▶ `nuucp` - the UNIX-to-UNIX system copy administration user
- ▶ `listen` - the Network Listener Daemon administration user

Use the **passmgmt** command, with the `-d` option, to delete accounts in `/etc/passwd` and `/etc/shadow`. Here is an example:

```
# passmgmt -d smtp
```

This command removes the `/etc/passwd` and `/etc/shadow` entries for `smtp`. The remaining system accounts (except the root account) should also be modified for added security.

System accounts listed in `/etc/passwd` have no shell listed. Those accounts also have an NP string (meaning “no password”) listed in the `/etc/shadow` file. By default, this is sufficient. However, some additional steps can be taken to add more security. Use the `-l` option of the `passwd` command to lock accounts. For example, to lock the `uucp` account use the command:

```
# passwd -l uucp
```

Also, use the **-e** option with the **passwd** command or edit the `/etc/passwd` file manually to change the default shell for those accounts to `/usr/bin/true`. For example:

```
# passwd -e uucp
Old shell: /sbin/sh
New shell: /usr/bin/true
```

This prevents an unauthorized user from using the `uucp` account to start a shell session on your system.

Note: The `/usr/bin/true` command is a UNIX command typically used in shell script to have a true condition.

7.2.5 Cron and at security

The “cron” and “at” services execute commands at a future time. User submission for the cron service is handled by the **crontab** command. The **at** and **batch** commands are used to submit jobs to the at service.

The cron and at services can be problematic since commands are executed at a future point in time. An attacker may use these systems to implement a “logic bomb” or other type of programmed attack that begins at some point in the future. It is better to restrict access to the at and cron systems to prevent attacks and abuse.

Cron configuration

The access control files are stored in the `/usr/lib/cron` directory:

- ▶ The `cron.deny` file
- ▶ The `cron.allow` file

The “allow” file is checked first to see if the account is explicitly allowed to use the system. If the file does not exist or the account is not listed in this file, the “deny” file is checked. If the account is explicitly listed in the “deny” file, then access is refused. Otherwise, access is permitted. If neither the “deny” nor the “allow” files exist, then only the root account can use the at or cron system. Add only the accounts that need access to the “allow” file.

At configuration

The `at.deny` and `at.allow` files are used to manage access to the at system. The same considerations that apply to the cron allow/deny files are valid for the **at** command.

Usage

The following command lists all the scheduled tasks for the current user:

```
# crontab -l
```

The Domino user administrator can use the **cron** and **at** Solaris commands to schedule maintenance tasks outside of Domino. For example, the administrator can schedule the Domino server shutdown and do some maintenance procedures, such as starting the compact, fixup or updball Domino programs on some databases.

7.2.6 The init system

The Solaris operating environment init system manages system services. Some services may not be needed or should be modified to strengthen the security posture of a system.

System services are started by the init system. There are some services that may allow a system to be compromised due to incorrect configuration. To disable services started by init, simply rename or delete the initialization script in the init system run level directory. The run level directories contain the scripts for starting or stopping services for the system run level. The system run level directories and their purposes are listed here:

- ▶ /etc/rcS.d single user
- ▶ /etc/rc0.d shutdown
- ▶ /etc/rc1.d start
- ▶ /etc/rc2.d multi-user
- ▶ /etc/rc3.d multi-user (default)
- ▶ /etc/rc4.d multi-user (unused)
- ▶ /etc/rc5.d shutdown and power off
- ▶ /etc/rc6.d shutdown and reboot

These directories contain initialization scripts to start or stop services. Initialization scripts that begin with either an “S” or a “K” are executed by the init system. S scripts *start* services, and K scripts *stop* or *kill* services. If you rename the scripts, make sure the name does not begin with these letters. Instead of deleting these files, we recommend placing an underscore character (_) at the beginning of the original file name. This makes it easy to enable services that may be needed later. For example:

```
# cd /etc/rc.2
# mv S99dtlogin _S99dtlogin
```

For security purposes, only required services should be enabled. The fewer services that are enabled, the less likely it is that an attacker will discover a way to exploit the system using an enabled service.

7.2.7 System default umask

The default system umask has changed to 022 from previous versions of Solaris. This value determines the default permissions for a created file; 022 is a good secure value. It can also be adjusted by altering the CMASK variable in the `/etc/default/init` file.

7.2.8 Log files

Log files are used by the system and applications to record actions, errors, warnings, and problems. They are often quite useful for investigating system quirks, for discovering the root causes of tricky problems, and for watching attackers.

Log type

There are typically two types of log files in the Solaris operating environment:

- ▶ System log files, which are typically managed by the syslog daemon
- ▶ Application logs, which are created by the application

It is possible to redirect some events of Lotus Domino in the syslog system file. See Appendix D, “Domino and syslog” on page 379 for details.

Syslog

The syslog daemon receives log messages from several sources and directs them to the appropriate location based on the configured facility and priority. The facility (or application type) and the priority are configured in the `/etc/syslog.conf` file to direct the log messages. The destination can be a log file, a network host, specific users, or all users logged in to the system.

By default, the Solaris operating environment defines two log files in the `/etc/syslog.conf` file. The `/var/adm/messages` log file contains a majority of the system messages. Save the file and use the following command to force syslogd to reread its configuration file:

```
# kill -HUP `cat /etc/syslog.pid`
```

All of these files should be examined regularly for errors, warnings, and signs of an attack. This task can be automated by using log analysis tools or a simple `grep` command.

Login trace

The `/var/adm/loginlog` file does not exist in the default of the Solaris operating environment installation, but it should be created. If this file exists, the login program records failed login attempts.

7.2.9 The login command

The `login` command is part of the authentication process to access a local Solaris operating environment account. It is used on the console and by the `in.telnetd` daemon to determine if a user may be granted access to the system.

By default, only the root user can log into a Solaris operating environment system from the console device. The console device is defined by the following entry in the `/etc/default/login` file:

```
CONSOLE=/dev/console
```

When this line is commented out, the root account can log directly into the system over the network via telnet, in addition to the console. This is not secure and should be avoided. Do not alter the default configuration.

7.3 Network service security

The Solaris operating environment is designed to provide customers with full access to most network services by default. This allows administrators and users to install and configure the Solaris operating environment systems as quickly as possible. Customers are encouraged to disable all unnecessary services for performance and security reasons.

There are many possible ways to attack network services. Network weakness can be introduced by many factors, including the following:

- ▶ Programming flaws
- ▶ Using weak authentication
- ▶ Transferring sensitive data in unencrypted format
- ▶ Allowing connections from any network host

These weaknesses allow a system to be compromised by an attacker. In addition to vigilance in ridding your system of these problems, safer network service alternatives should be used whenever possible.

Sun has a product for secure network services, based on Kerberos, called the Sun Enterprise Authentication Mechanism (SEAM product). Kerberos is a centralized network security architecture that uses symmetric encryption and a ticket mechanism to provide strong authentication. The SEAM product also uses strong encryption.

There is also a tool known as Secure Shell (SSH) that provides strong authentication and encryption capabilities. There are both commercial and open source versions available.

Access control can be provided, thereby configuring network services to only handle connections from approved systems. Wietse Venema's TCPWrapper toolkit provides access control and additional security checks. It manages TCP-based services managed from inetd.

Additional information and details on how to obtain these tools are given in 7.4, "Security tools" on page 184

7.3.1 Telnet

Telnet is a user-interactive service used to log into and access a remote system on the network. Unfortunately, this service provides little in the way of security. The only authentication information required is user name and password. Neither of these pieces of information is encrypted while in transit, so telnet service is vulnerable to a variety of attacks including:

- ▶ Man in the middle attack
- ▶ Session hijacking
- ▶ Network sniffing

If you must use a telnet daemon which does not support encryption, TCPWrappers can be used to limit the hosts which may connect to a system. By restricting access to services based on IP addresses, a system can limit its exposure to network attacks.

7.3.2 Remote access services (rsh, rlogin, and rcp)

The default authentication mechanism of the r* daemons (r* is an abbreviation for remote commands) uses the IP address of a system in combination with the userid for authentication. No additional authentication is required. Considering the ease with which an IP address and userid may be stolen or misused, this is clearly not a secure mechanism.

The r* commands should never be used in this manner and no servers should offer the service in this manner.

Secure Shell (SSH) can be used to improve the security of the `r*` commands.

7.3.3 Remote execution service (rexec)

The remote execution server daemon, `in.rexecd`, is started from the file `/etc/inetd.conf` when a connection request is made. This daemon provides remote execution facilities based on user name and password information.

Once authenticated, the daemon executes the command passed along with the authentication information. As with the `in.telnetd` daemon, neither the user name nor password is encrypted while transmitted over the network.

This exposes the `in.rexecd` daemon to the same man in the middle, session hijacking, and network sniffing attacks as the `in.telnetd` and `in.ftpd` daemons. For this reason the `in.rexecd` entries in `/etc/inetd.conf` file should be removed.

7.3.4 FTP

The FTP daemon has many of the same problems as the telnet daemon. All authentication information transmitted over the network is in clear text, in much the same fashion as the telnet protocol. This exposes the ftp protocol to many of the same attack scenarios as telnet, including man in the middle, session hijacking, and network sniffing.

If you must use FTP, you have to configure the `/etc/ftpusers` file which is used to restrict access to the system through FTP. All accounts not allowed to use the incoming FTP service should be specified in this file. At a minimum, this should include all system accounts (i.e., `bin`, `uucp`, `smtp`, `sys`, and so forth) in addition to the root account.

Another service, the trivial FTP service (`in.tftpd`), exists to provide diskless systems with a way to get files on the network. This is less secure than even FTP. By default, it is not enabled in the Solaris operating system.

7.3.5 Managed services: inetd

The `inetd` service manages many of the minor network services available on a system. Its configuration file, `/etc/inetd.conf`, defines its operation. An ideal secured server should not have an `/etc/inetd.conf` or run `inetd`, as the daemons started in the `/etc/inetd.conf` are frequently not needed.

To disable a service, edit the `/etc/inetd.conf` file and place a comment character (`#`) in front of the line containing the service definition. Once this is completed, send a HUP signal to the `inetd` process, using the `kill` command as follows:


```
# kill -s HUP pid
```

where pid is the Process ID of the inetd process.

This will cause it to reread its configuration file.

Of the daemons started from the `/etc/inetd.conf`, the remote access services, FTP, TFTP, and TELNET services have already been discussed. The remaining `/etc/inetd.conf` entries should be removed. The only one that can be left is the `netstat`, a service that provides a list of current network connections, useful for troubleshooting network issues.

Once removed, the server `inetd` should be restarted and applications tested to verify that required functionality has not been affected.

7.3.6 Network File System (NFS)

The Network File System (NFS) permits the sharing of file systems on network-connected machines. A Solaris operating environment system can be either an NFS server, NFS client, both, or neither.

From a security perspective, the best option is to not provide NFS services or accept them from any other systems. To disable all client and server NFS daemons, the following startup scripts should be disabled on the system:

- ▶ `/etc/rc1.d/K65nfs.server`
- ▶ `/etc/rc1.d/K80nfs.client`
- ▶ `/etc/rc2.d/S73nfs.client`
- ▶ `/etc/rc2.d/K60nfs.server`
- ▶ `/etc/rc3.d/S15nfs.server`

7.3.7 Automount

The automount service manages automated NFS mounts. The automount utility installs `autofs` mount points and associates an automount map with each mount point. The `autofs` kernel module monitors attempts to access these mount points.

Ideally, it should be disabled since not only does it run as a privileged daemon, but it also uses NFS and RPC. The automounter can be disabled by renaming the `/etc/rc2.d/S74autofs` file.

7.3.8 Sendmail

The sendmail daemon is used on a Solaris operating environment system both to forward and to receive mail from other systems. A more secure Mail Transport Agent (MTA) should be used.

The sendmail daemon has been subject to numerous denial of service attacks from the Internet.

To disable the sendmail daemon, rename the sendmail startup scripts `/etc/rc2.d/S88sendmail` and the stop script `/etc/rc1.d/K57sendmail` too.

7.3.9 Print services

When a Solaris operating environment system is installed using the End User, Development, or Entire Distribution cluster, the line printing packages are installed.

The Lotus Domino client is no longer available in R5 for UNIX platforms, so we don't need to have this service running on the server.

To disable it, you should remove the following line from the `/etc/inetd.conf` file:

```
printer stream tcp nowait root /usr/lib/print/in.lpd in.lpd
```

The `in.lpd` entry should also be removed from the `/etc/inetd.conf` file.

7.3.10 Reducing inetsvc

The `inetsvc` file contains all the network services the Solaris system will start at the OS boot.

Based on the recommendations made in this section, it is possible to construct a minimized `/etc/init.d/inetsvc` file which contains only the essential components.

By commenting out all of these entries, the resulting script should look like the following:

```
#!/bin/sh
/usr/sbin/ifconfig -au netmask + broadcast +
/usr/sbin/inetd -s -t
```

With this minimal file you will have a well secured system.

7.3.11 The Solaris **ndd** command

You can increase the Network Secure level of your Solaris system using the Solaris **ndd** command. It is used to examine and set kernel parameters, namely the TCP/IP drivers.

Usage

Most kernel parameters accessible through **ndd** can be adjusted without rebooting the system. To see which parameters are available use the following **ndd** commands:

- ▶ # **ndd /dev/arp \?**
- ▶ # **ndd /dev/icmp \?**
- ▶ # **ndd /dev/ip \?**
- ▶ # **ndd /dev/tcp \?**

These commands list the parameters for the ARP, IP, ICMP, and TCP drivers. “\?” will list all parameters for the driver and indicate whether the parameter is read only, write only, or read and write. The current parameter value or status information can be read by specifying the driver and parameter names.

This example shows the output of an **ndd** command examining the debugging status of the ARP driver. (The output “0” indicates that the option is disabled.)

```
# ndd /dev/arp arp_debug  
0
```

Parameter values specified by **ndd** are integers with “0” meaning disable, “1” meaning enable, or a large integer to set a time or size value.

Basic recommendations

Setting parameters requires the “-set” option, the driver name, the parameter name, and the new value.

For example, to enable debugging mode in the ARP driver use the following **ndd** command:

```
# ndd -set /dev/arp arp_debug 1
```

One security use for **ndd** is to disable the IP forwarding, the process of routing packets between network interfaces on one system. Systems that allow packet forwarding are targets for attackers, as they provide access to other systems and networks.

Packet forwarding is easily disabled on a Solaris system. Simply creating a file named `/etc/notrouter` will disable IP forwarding at boot time.

IP forwarding can also be switched on or off while the system is operating, using the `ndd` command. Use this command to disable IP forwarding:

```
# ndd -set /dev/ip ip_forwarding 0
```

To view the current IP forwarding table, use the following command:

```
# ndd /dev/ip ip_ire_status
```

7.4 Security tools

A lot of security tools can be found over the Internet. In this section we describe some of the more common ones. Always try to use the latest available versions of these tools.

7.4.1 The `sudo` tool

Sudo (superuser do) allows a system administrator to give certain users (or groups of users) the ability to run some (or all) commands as root while logging all commands and arguments.

Sudo is free software and is distributed under a BSD-style license. It is currently maintained by Todd Miller, and you can find it at:

<http://www.courtesan.com/sudo/>

7.4.2 TCP wrappers

With this package you can monitor and filter incoming requests for SYSTAT, FINGER, FTP, TELNET, RLOGIN, RSH, EXEC, TFTP, TALK, and other network services. The package provides tiny daemon wrapper programs that can be installed without any changes to existing software or existing configuration files. The wrappers report the name of the client host and of the requested service; the wrappers do not exchange information with the client or server applications, and impose no overhead on the actual conversation between the client and server applications.

For more details see the URL:

<ftp://ftp.porcupine.org/pub/security/index.html>

7.4.3 Secure shell (ssh)

Ssh is a program used to log into another computer over a network, to execute commands in a remote machine, and to move files from one machine to another. It provides strong authentication and secure communications over unsecured channels. It is intended as a replacement for rlogin, rsh, and rcp.

You can find it at:

<ftp://ftp.gw.com/pub/unix/ssh/>

7.4.4 Titan

Titan is a collection of programs, each of which either fixes or tightens one or more potential security problems with a particular aspect in the setup or configuration of a UNIX system. Conceived and created by Brad Powell, it was written in Bourne shell, and its simple modular design makes it trivial for anyone who can write a shell script or program to add to it, as well as completely understand the internal workings of the system.

Titan does not replace other security tools, but when used in combination with them it can help make the transformation of a new, out-of-the-box system into a firewall or security-conscious system into a significantly easier task. In a nutshell, it attempts to help improve the security of the system it runs on.

You can find it at the Internet site:

<http://www.fish.com/titan/>

7.4.5 The fix-modes script

You can find Casper Dik's fix-modes script fixes, described previously, at:

<ftp://ftp.wins.uva.nl/pub/solaris/fix-modes.tar.gz>

7.4.6 The SANS scripts

The SANS community (System Administration, Networking, and Security) has been developing automated tools, called scripts, to remove vulnerable unneeded services, tighten loose settings and make other changes to improve security on Internet-connected computers. There is also a version for the Solaris system.

You can find more information on the SANS Web site at:

<http://www.sans.org>

7.4.7 The logcheck Perl script

Logcheck, which uses Perl, is a software package that is designed to automatically run and check system log files for security violations and unusual activity.

You can find logcheck information at:

<http://www.psionic.com/abacus/logcheck/>

More information on the latest version of Perl is at:

<http://www.perl.org/>

7.5 Domino and Notes security basics

In this section, we present a short overview of Domino and Notes security issues.

For detailed information, see *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341-00.

7.5.1 Notes certificates

A Notes certificate is an electronic stamp, like a stamp on a passport, which verifies to the server that you are who you say you are. In actuality, a certificate is a digitally signed message added by a certifier to a Notes ID file.

When you register users and servers, Domino automatically creates a Notes certificate for each user and server ID.

You can also create Internet certificates for user IDs using a Domino or third-party certificate authority (CA). Domino creates Internet certificates using the X.509 certificate format. For more information on Internet certificates, see 9.3.1, “Internet certificates” on page 233.

Note: The certificate itself does not contain any private information; it is therefore open to the public and can be distributed anywhere.

7.5.2 Certification hierarchies

Lotus Notes used to provide two types of certification: flat and hierarchical. Organizations using flat names may use several certifier IDs. Each user ID and server ID can include separate certificates generated by each flat certifier ID. Organizations using hierarchical certification have one organization certifier and, optionally, up to four layers of organizational unit certifiers below.

Flat certificates

Before the introduction of hierarchical certificates, flat certificates were the only way to register users and servers. They are supported in Lotus Notes/Domino R5.0 only for compatibility with earlier releases.

New installations are encouraged to start with hierarchical names, and existing flat installations are encouraged to convert to hierarchical, because of the increased security and flexibility of access control, ID file generation and certification, and maintenance.

Hierarchical certificates

In hierarchical certification, an organization may be layered with the organization certifier at the top and up to four layers of organizational unit certifiers below. When users or servers are registered with a certifier, they receive a certificate signed by that certifier and inherit the certification hierarchy of the layers above.

Users and servers may authenticate with each other if they have at least one common ancestral certificate. Entities that don't share at least one common ancestor can still authenticate by going through a cross-certification process.

Cross-certification

Cross-certificates are used to allow users and servers from different hierarchically-certified organizations to access servers in other organizations and to verify the digital signature of a user from another organization. Servers store cross-certificates in the Domino Directory. To access servers, users store cross-certificates in their Personal Address Books.

7.5.3 Notes IDs

Domino uses ID files to identify users and to control access to servers. Every Domino server, Notes certifier, and Notes user must have an ID. When an administrator registers users and servers, Domino automatically creates their IDs.

Important: The security of the entire Domino system relies heavily on the secure creation, distribution, administration, and archiving of certificates and Notes IDs or their ID files so that they cannot be compromised. Store ID files on secure media and keep them safe.

Contents of a Notes ID

After the registration process, the ID file contains:

- The user's name and Notes license number

- ▶ Two public and private key pairs
- ▶ Two certificates for the user
- ▶ A certificate for each ancestor certifier

Password protection

ID files should be password-protected to avoid unauthorized use. When you password-protect your ID, a key that is derived from the password encrypts the data on the ID. Then, when you attempt to access mail, open a server-based database, or examine ID file information, you are prompted to enter a password.

7.5.4 Notes validation and authentication

Whenever a Notes client (or Domino server) attempts to communicate with a Domino server for replication, mail routing, or database access, two security procedures use information on the client's ID to verify that the client is legitimate: validation and authentication.

1. Validation establishes trust of the client's public key. If validation occurs successfully, authentication begins.
2. Authentication verifies the identity of the user. Authentication uses the public and private keys of the client and the server in a challenge/response interaction.

7.6 Protecting a Domino server

You as an administrator have to ensure that all data on your Domino server is protected. From the very beginning of server setup on, you should always keep security in mind. Basic security settings have to be made as soon as a user or server gains access to the server on the network. If you set up servers for Internet or intranet access, you have to set up additional server security. In addition, set up a firewall server to protect Internet servers from unauthorized access from outside the organization's network.

7.6.1 Protecting access during Domino server setup

The httpsetup task can be loaded on the server to provide access for an administrator to perform server setup, since there is no Notes client for Solaris that can be executed locally on the machine.

Once the setup forms have been completed in the setupweb.nsf database with details such as domain, certifier name, server name, and administrator name, they are submitted to the server for the setup task to process.

Note: The field details from the form are sent in clear text format to the server; therefore, ID names, and password details are vulnerable during this time.

The administrator may use FTP to retrieve copies of newly created certifier, server, or administrator ID files. Using an unsecured FTP session means that the contents of the ID files are vulnerable to network sniffers during transit between the server and the receiving machine.

7.6.2 Setting up basic Domino server security

You can use server documents in the Domino Directory to control access to a Domino server. In addition, you can restrict the activities that users and servers may perform on the server.

Defining server administrators

Define server administrators by setting the “Administrators” field on the Basics tab. Only administrators who are listed in this field can use the console from the Domino Administrator client.

Deny anonymous server access

You can give server access to Notes users and Domino servers outside of the organization without issuing a cross-certificate. In addition, you can preserve user anonymity since no user name is recorded, for example, in the log file (Log.nsf) or in the User activity dialog box. Set the “Allow anonymous Notes connections” field on the Security tab to “No” to deny anonymous server access.

Secure the server console

To avoid unauthorized access to your server console, always run the server in the background using a shell script, as described previously (see Chapter 3, “Installing Domino R5 on Sun Solaris” on page 43). A console log can be displayed for monitoring by using shell script features to tail the log file.

Allow or deny access to your server

Specify which Notes users and Domino servers are authorized to access the server. Set the following fields on the Security tab:

- ▶ Access server
- ▶ Not access server

Select whether only users listed in the current Domino Directory can access the server or not.

Tip: Use groups instead of single user names. Groups are easier to administer.

Allow access to create new databases or replicas

Specify which Notes users and Domino servers are authorized to create databases and replica databases on the server. Set the following fields on the Security tab:

- ▶ Create new databases
- ▶ Create replica databases

Allow or deny use of monitors

Specify which Notes users and Domino servers are authorized to use monitors. You can use monitors to track server resources and network and system activity. Set the following fields on the Security tab:

- ▶ Allowed to use monitors
- ▶ Not allowed to use monitors

Restrict access to the Web Administrator

Specify which Internet or intranet user can use the Web Administrator to administer the server from a browser. Set the “Administer the server from a browser” field on the Security tab.

Restrict server agents

Specify which Notes users and Domino servers are allowed to run which kinds of agents on the server. Set the following fields on the Security tab:

- ▶ Run personal agents
- ▶ Run restricted LotusScript/Java agents
- ▶ Run unrestricted LotusScript/Java agents

Controlling the level of authentication for Web clients

Specify what type of names will be accepted during name and password authentication. Set the “Web server authentication” field on the Security tab to “Fewer name variations with higher security.” By restricting the number of available entries, you will ensure that a hacker is less likely to find a match by guessing a user name.

Restrict passthru access

Specify which Notes users and Domino servers can access the server as a passthru server and specify the destinations they may access. You can do that by setting the following fields on the Security tab:

- ▶ Access this server
- ▶ Route through
- ▶ Cause calling
- ▶ Destinations allowed

Restrict server access by browser users running Java or JavaScript programs

Specify which Web browser users can run Java or JavaScript programs on the server. Set the following fields on the Security tab:

- ▶ Run restricted Java/Javascript/COM
- ▶ Run unrestricted Java/Javascript/COM

Figure 7-1 shows the Security tab on the Server document in the Domino Directory.

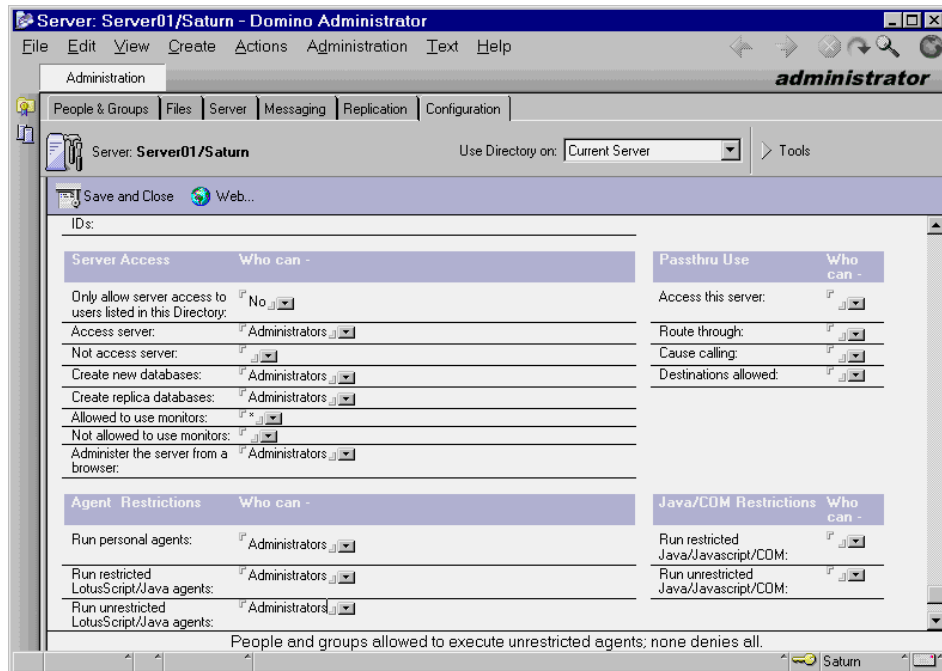


Figure 7-1 Server Document Security tab

7.6.3 Setting up additional Domino server security

You can tighten Domino Server security, especially the security of your Web server, by setting additional parameters.

Restrict Web browser access to the ?OpenServer URL parameter

Specify whether browser users can see a list of all databases on the server. By default, users cannot display a list of databases even if they have access to the server. Set the “Allow HTTP clients to browse databases” field on the Internet Protocols - HTTP tab to “No.”

Control Web browser access to files on their server’s hard drive

Specify who can access files (for example, HTML, GIF or JPEG) on a server’s hard drive. For more information, see 9.3.6, “Domino File Protection” on page 238.

Secure the server with SSL

Set up SSL security for Internet and intranet users to authenticate the server, encrypt data, prevent message tampering, and, optionally, authenticate clients. For more information, see “Invoking SSL on Your Domino Server” in Chapter 4 of *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341-00.

Secure the server with name-and-password authentication

When using name-and-password authentication, also known as basic password authentication, Domino asks for a name and password only when an Internet or intranet client tries to access a database on the server. Set up basic password authentication using the following steps:

1. Create a Person document for each user in the Domino Directory on the Domino server.
2. Assign an Internet password to each user.
3. Specify which Internet protocols require a name and password. Do this by setting the appropriate fields on the Ports - Internet Ports tab in the Server document.
4. Set the database ACLs.

Note: You can use basic password authentication with either TCP/IP or SSL on any servers that run an Internet protocol—namely, NNTP, LDAP, POP3, HTTP, SMTP, IIOP, or IMAP.

Secure the server with session-based authentication

If you are setting up name-and-password authentication for an HTTP server, you have an additional method for name-and-password authentication: session-based authentication. Session-based name-and-password authentication offers greater control over user interaction than basic name-and-password authentication, and lets you customize the form in which users enter their name and password information. It also allows users to log out of the session without closing the browser.

Set up session-based authentication by completing the following fields on the Internet Protocols - Domino Web Engine tab in the Server document:

- ▶ Session authentication
- ▶ Idle session timeout
- ▶ Maximum active sessions

For more information, see “Setting up session-based name-and-password authentication” in the security section of Domino 5 Administration online help.

Allow anonymous Internet and intranet client access

Determine whether Internet and intranet users are allowed to access the server anonymously. Use the following steps to do this:

1. Set the “Anonymous” field for the protocol that you want accessed anonymously on the Ports - Internet Ports tab in the Server document.
2. Include “Anonymous” in the ACL of the database for which you want to give free, anonymous access.

Authenticate Web clients using a secondary Domino directory or LDAP directory

If your organization uses a secondary Domino directory or an LDAP directory to verify client certificates, you can set up Domino to check those additional directories. To do so, set up the secondary Domino and LDAP directories as trusted domains in the Directory Assistance database.

For more information, see “Authenticating Web SSL clients in secondary Domino and LDAP directories” in the security section of Domino 5 Administration online help.

Authenticate Web clients for a specific realm

Allow Web users to access a certain drive, directory, or file on a Domino server and prevent Domino from prompting users for a name and password for different realms. Setting a realm for specific applications also protects the Domino server databases and other application directories from unauthenticated use. When a

user accesses a page on a Domino Web site, the browser keeps track of user credentials, based on the realm that the Domino server sends to the browser. A realm is a string, which is typically a URL path, that the server sends to indicate the location, or path, for which the user has been authenticated.

For example, if your server name is `www.acme.com`, then `www.acme.com` is the top-level realm and `www.acme.com/doc`, `www.acme.com/hr`, and `www.acme.com/marketing` are the lower-level realms. If a user authenticates with the server when accessing the home page for `www.acme.com`, then the user is authenticated for `www.acme.com` and all lower-level realms.

However, if the user accesses `www.acme.com/doc` first, enters a name and password and is authenticated, and then accesses `www.acme.com/hr`, Domino prompts the user for credentials again. This second prompt occurs because the browser examines the list of realms for which Domino has successfully authenticated the user and finds `www.acme.com/doc` in the browser realm list. Since `www.acme.com/hr` is not a subdirectory of `www.acme.com/doc`, Domino requires the user to enter credentials again.

To prevent users from being prompted multiple times for their names and passwords, direct them to access and authenticate with the highest level realm that they need to access. This way, Domino asks users for their credentials only once during the browser session.

You protect server files from Web access by creating a File Protection document in the Domino Directory.

You specify Web realms by creating Web realm documents in the Domino Directory.

For more information, see 9.3.5, “Domino Web Realms” on page 237, or “Controlling Web browser access to server files” in the security section of Domino 5 Administration online help.

7.7 Setting up Domino database security

Once the actual server access and connections to the server have been restricted, access to the data in the databases and applications on the server must be considered.

Tip: To ensure that all security mechanisms are covered, disable all access and then re-enable only the specific areas of access that are known to be required. Having to remember to enable a feature is much less of a risk than having to remember to disable one, as security may already have been breached.

7.7.1 Review database ACLs

Every database has an access control list (ACL) that specifies the level of access that users and servers have to the database. The access levels are the same for users and servers. Access levels assigned to users determine the tasks that users can perform, while those assigned to servers determine what information within the database the servers can replicate.

You must have Manager access to modify the ACL.

The following areas of database access control should be reviewed and set for all databases and templates on the server:

- ▶ Enter an Anonymous entry in the ACL with the appropriate access level.
- ▶ Set the Default entry to No Access.
- ▶ Check appropriate use of Read public documents and Write public documents.
- ▶ Set an appropriate Maximum Internet Name & Password access level.
- ▶ Use groups instead of single user entries. Groups prevent ACLs from becoming too big and unmanageable.
- ▶ Assign user types since they provide an additional level of security.
- ▶ Consider the use of consistent ACLs.

For more information on ACLs, see “The database access control list” in the security section of Domino 5 Administration online help.

Figure 7-2 on page 196 shows the Access Control List window.

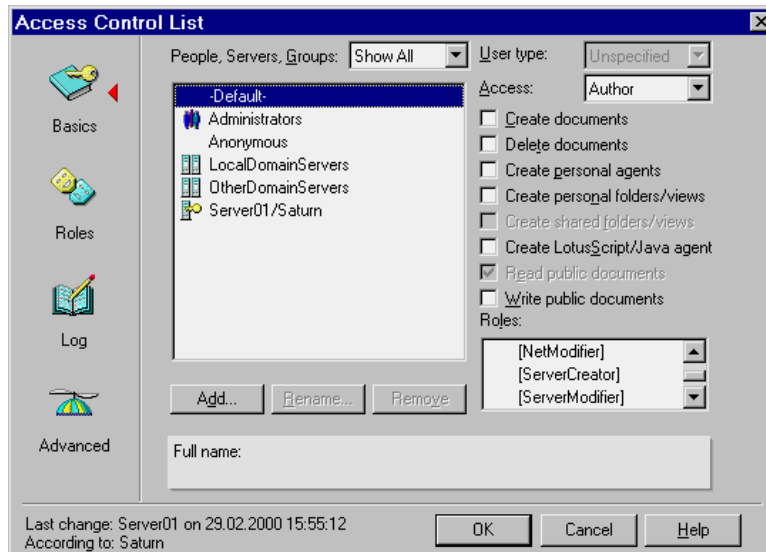


Figure 7-2 Database Access Control List

7.7.2 Consistent ACLs

You can ensure that an ACL remains identical on all database replicas on servers, as well as on all local replicas that users make on workstations or laptops, by selecting "Enforce a consistent Access Control List" in the "Advanced" section of a database ACL.

You need Manager access to change an ACL. Select the "Enforce a consistent Access Control List" setting on a replica whose server has Manager access to other replicas to keep the access control list the same across all server replicas of a database. If you select a replica whose server does not have Manager access to other replicas, replication will fail because the server has inadequate access to replicate the access control list.

Manager access is awarded to you by default when a database is local because security doesn't need to be enforced. However, if the "Enforce a consistent access control list" option is selected, your local rights can never exceed the rights you have been assigned on the server. If you have been assigned Author access and the option is enabled, you will have Author access locally, not Manager access.

If a user changes a local or remote server database replica's ACL when the "Enforce a consistent access control list" option is selected, the database stops replicating. The log file records a message indicating that replication could not proceed because the program could not maintain a uniform access control list on replicas.

Enforcing a consistent access control list does not provide additional security for local replicas. To keep data in local replicas secure, encrypt the database. For information on encryption, see "Database encryption" in the Domino 5 Administration online help.

7.8 Anti-virus products for Domino

A number of anti-virus products are available for Domino R5 running on Solaris. These products are best deployed on the mail and replication gateways, and firewall servers. With this architecture the performance impact of running virus detection is localized to the key gateway servers. The current products include Norton AntiVirus, by Symantec, and ScanMail, by Trend Micro.

This section provides a brief overview of these two products.

7.8.1 Norton AntiVirus

At the time of this writing, the current version of Norton AntiVirus (NAV) Lotus/Notes Domino is version 2.1. It requires the use of Domino R5.0.4a or higher and Solaris 2.6, 7 or 8.

Norton AntiVirus is well integrated into the Lotus Notes environment. All scanning is configured and initiated from the NAV Settings database. All reports and virus dispositions are handled through the NAV Log database.

Norton AntiVirus for Lotus Notes/Domino for Solaris SPARC can be configured to do any of the following:

- ▶ Eliminate viruses automatically on detection.
- ▶ Quarantine infected documents and email for administrator review.
- ▶ Delete infected items.

Norton AntiVirus runs as an addin Domino server task. After installation, you can find the NNTASK in the ServerTask line of the Notes.ini file.

You can view and manage some Norton AntiVirus operations directly from the Domino server console window. Use the following syntax at the command prompt:

```
tell nav <command>
```

To get a complete list of the possible commands, use:

```
tell nav help
```

Norton AntiVirus relies on up-to-date information to detect and eliminate viruses. Symantec regularly supplies updated virus definition files, which contain the necessary information about all newly discovered viruses.

Using LiveUpdate, Norton AntiVirus connects automatically to special Symantec sites and determines if your virus definitions need updating. If so, it downloads the proper files and installs them in the proper location.

For more information on this product visit the Symantec Web site at:

<http://www.symantec.com>

7.8.2 ScanMail

At the time of this writing, version 2.5 of Scan Mail for Lotus Notes (SMLN) for Solaris is available for Lotus Domino R4 and 5.

Once ScanMail is installed, it scans old message attachments in mailboxes and databases to root out any old infections. Thereafter, all mail is scanned in real time, and mail with multiple recipients does not have to be scanned more than once. ScanMail for Lotus Notes is supporting all types of shared mail and has implemented a very efficient in-memory scan and incremental scan of databases to increase the scan performance. The server tasks of SMLN are fully integrated into the Lotus Domino Administration client.

If an infection is found, a customizable alert message can be sent to administrators, the sender, and the receiver. ScanMail for Lotus Notes has the capability to send different warning messages for internal and external mail recipients and supports, as well, a rich text notification for different character sets. Infected files can also be automatically cleaned and sent to their recipients with no disruption in message delivery.

ScanMail for Lotus Notes is enhanced with an interface to create and manage e-mail filter rules. Using these rules, it is possible to manage the size, content, delivery time, and blocking of unwanted e-mail by sender or recipients. It also supports attachment blocking based on the header information of the attachment.

ScanMail maintains a comprehensive and robust virus incident activity log in Notes database format, which may be viewed by:

- Incident date

- ▶ User name
- ▶ Virus name (if known)
- ▶ Service type (email, database or replication)
- ▶ Action taken (cleaned, moved, deleted, passed)

For more information on this product visit the Trend Web site at:

<http://www.antivirus.com>

7.8.3 Other antivirus products

Sybari Antigen for Lotus Domino protects Domino systems from viruses through three components. NShield enables real-time protection of Notes databases, NWall scans the incoming and outgoing e-mail streams in real time, and NScan provides scheduled scans of Notes databases. For more information see <http://www.sybari.com/>

7.9 Summary

In this chapter we discussed how to implement a secure system for Domino running on Solaris. We considered this issue from a Solaris operating environment point of view and from a Domino application point of view.



Domino Directory services

In this chapter we discuss Domino Directory services.

In addition to the Domino Directory itself, Domino provides three directory service features: Directory Catalog, Directory Assistance, and the LDAP service. These features help users find user names, e-mail addresses, and other information in the Domino Directory.

The Directory Catalog consolidates key information about users and groups from one or more Domino Directories into a small, lightweight database. Lotus Notes users can use a local replica of the Directory Catalog—a Mobile Directory Catalog—to quickly address mail to users throughout the organization, even if the organization uses a large directory or multiple directories. In organizations with multiple Domino Directories, a Directory Catalog on a server combines these directories into a single database so that a server can look up names in one database rather than in multiple Domino Directories.

Directory Assistance is a feature that helps manage user name lookups in organizations that use multiple Domino Directories, alone or in combination with third-party LDAP directories. A Directory Assistance database associates each Domino or LDAP directory with specific hierarchical names so that when looking up a name, Domino first searches the directory that contains names in that hierarchy.

Extended Directory Catalog combines advantages of the Domino Directory and the Directory Catalog by aggregating entries from multiple Domino directories into a single directory database.

You can set up a Domino server to run the Lightweight Directory Access Protocol (LDAP) service to enable LDAP clients to search for and modify information in the Domino Directory. The Domino LDAP service is LDAP V3 compliant.

8.1 The Domino Directory

The Domino Directory is the heart of the Lotus Domino architecture. Because of this, it is important to understand what it is and how it works. For a deeper understanding of the directory capabilities of the Domino directory see *Getting the Most From Your Domino Directory*, SG24-5986. Among the pertinent features of the Domino Directory are the characteristics in the following list. The Domino Directory is:

- ▶ A database that is automatically created when you set up the first Domino server in a domain. The file name is names.nsf.
- ▶ Replicated to all Domino servers in a domain, which means that all servers operate with the same Domino Directory.
- ▶ Automatically replicated to a new server that is added to a domain.
- ▶ The central registration and configuration database in a domain.
- ▶ A directory of information about users, servers, groups, and other objects that you might include in the directory yourself—for example, printers.
- ▶ A database that administrators use to manage the Domino system. For example, administrators create documents in the Domino Directory to connect servers for replication or mail routing, to register users and servers, to schedule server tasks, and so on.
- ▶ Referred to as the Public Address Book (PAB) or Name and Address Book (NAB) in previous releases of Domino and Notes.

Important: The Lotus Domino infrastructure relies heavily on a correctly configured Domino Directory. Changing or deleting documents may have serious consequences on the operation of the Domino environment. Therefore, safeguarding the Domino Directory from unauthorized access is vital.

8.1.1 Documents in the Domino Directory

The Domino Directory contains documents that control directory services, manage server tasks, and define server-to-server communication, among other things. Domino automatically creates some documents when you perform certain administrative tasks. For example, Domino creates a new Person document when you register a user. The table below explains the different types of documents in the Domino Directory.

Table 8-1 Document types stored in the Domino Directory

Document	Description
Certificate	Describes a certifier ID, including public key information
Configuration Settings	Configures mail, LDAP, and the Notes.ini file
Connection	Provides server and domain information for connecting servers for mail routing, replication, and newsfeeds
Domain	Defines a domain used in mail routing: Foreign, Non-adjacent, Adjacent, Foreign X.400, Foreign SMTP, Foreign cc:Mail, Global
External Domain Network Information	Contains names and addresses of servers in a secondary domain; allows Notes clients to connect to servers in the secondary domain
Group	Defines a list of users and servers for use in mail addressing, ACLs, and server access lists
Holiday	Defines holiday documents that users can download to their calendars
Location	Contains communication and other location-specific settings for use from a client
Mail-In Database	Defines the location and properties of a database that can receive mail
Person	Describes a user (Notes or non-Notes) in the directory
Program	Schedules Domino server tasks and other programs to run at specific times
Resource	Defines a resource that Notes clients can reserve by using the calendaring and scheduling feature
Server	Specifies server configuration settings, including server name, cluster name, security methods, ports, server tasks, Internet protocol, MTA, transactional logging, and so on

Table 8-1 Document types stored in the Domino Directory

User Setup Profile	Defines a standard set of configuration options for Notes clients, including connections, server accounts, replicas, bookmarks, and so on
File Identification	Verifies or associates a specific MIME type with an application.
Aggregation Configuration	Specifies configuration settings for extended directory catalog.

8.2 The Directory Catalog

A Directory Catalog consolidates entries for users, groups, mail-in databases, and resources from one or more Domino directories into a single, light-weight, quick-access database.

Typically, you create a server Directory Catalog and a Mobile Directory Catalog.

8.2.1 Server Directory Catalog

You create a server Directory Catalog so that Domino servers can search one database to find names in multiple Domino Directories.

Benefits of the server Directory Catalog include the following:

- ▶ When users are connected to the network, directory searches are done against the Directory Catalog instead of the multiple Domino directories that exist on the server. This also reduces network traffic and open database sessions.
- ▶ This single Directory Catalog database is significantly smaller than all the other Domino directories combined.
- ▶ Administrators can choose the attributes included in the Directory Catalog, and can create multiple Directory Catalogs with different content or sort orders.
- ▶ The Directory Catalog is fully LDAP-enabled and can be searched using standard LDAP clients.

8.2.2 Mobile Directory Catalog

You can use a User Setup Profile to create a replica of a Directory Catalog, known as a Mobile Directory Catalog, on Notes clients so users can quickly address mail to anyone in your organization, even when the senders are disconnected from the network.

Notes users can also create the Mobile Directory Catalog manually for themselves.

A User Setup Profile defines a standard set of configuration options for Notes clients, including connections, server accounts, replicas, bookmarks, and so on.

Type-ahead addressing searches the Mobile Directory Catalog rather than Domino directories on a server. This reduces network traffic. When type-ahead is activated, every time you type an address manually, Notes displays the first name it finds that matches the letters you type so that you can select a name rather than type the entire name.

8.2.3 Directory Catalog size

A Directory Catalog can combine entries from many Domino directories and still be very small. For example, several Domino directories that together contain more than 350,000 users and total 3 GB in size, when combined in a Directory Catalog, are likely to be only about 50 MB. In general, each entry in the Directory Catalog is slightly more than 100 bytes.

Note: The Directory Catalog and Mobile Directory Catalog are R5 features and do not work with R4.x Domino servers or Notes clients.

8.2.4 Setting up a Directory Catalog

To set up either a server Directory Catalog or a Mobile Directory Catalog complete these procedures:

1. Prepare a server for the source Directory Catalog.
2. Create and configure the source Directory Catalog.
3. Run the **dircat** task.
4. Do one of the following:
 - a. If you configured the Directory Catalog for server use, set up the Directory Catalog on servers.
 - b. If you configured the Directory Catalog for mobile use, set up the Mobile Directory Catalog on Notes clients.

Prepare a server for the source Directory Catalog

A source Directory Catalog is the replica of a Directory Catalog that the Directory Cataloger—the Dircat task—initially populates and then continues to update when changes occur in the full secondary Domino directories.

Perform the following steps:

1. Create a replica of each secondary Domino Directory that you want to include in the Directory Catalog. Use unique file names for these replicas.
2. Set up replication between your source Directory Catalog server and the secondary Domino directory servers.

For more information, see “Preparing a server for a source Directory Catalog” in the “Domino Directories” section of Domino 5 Administration online help.

Create and configure the source directory catalog

Perform the following steps from the Domino Administrator client or Notes client to create and configure the source Directory Catalog:

1. Choose File -> Database -> New and enter the server name that you prepared for the source Directory Catalog.
2. Select the Directory Catalog template (dircat.ntf), and click OK.
3. Open your new Directory Catalog.
4. Choose Create -> Configuration from the menu to bring up the Directory Catalog Configuration document, as shown in Figure 8-1 on page 207.

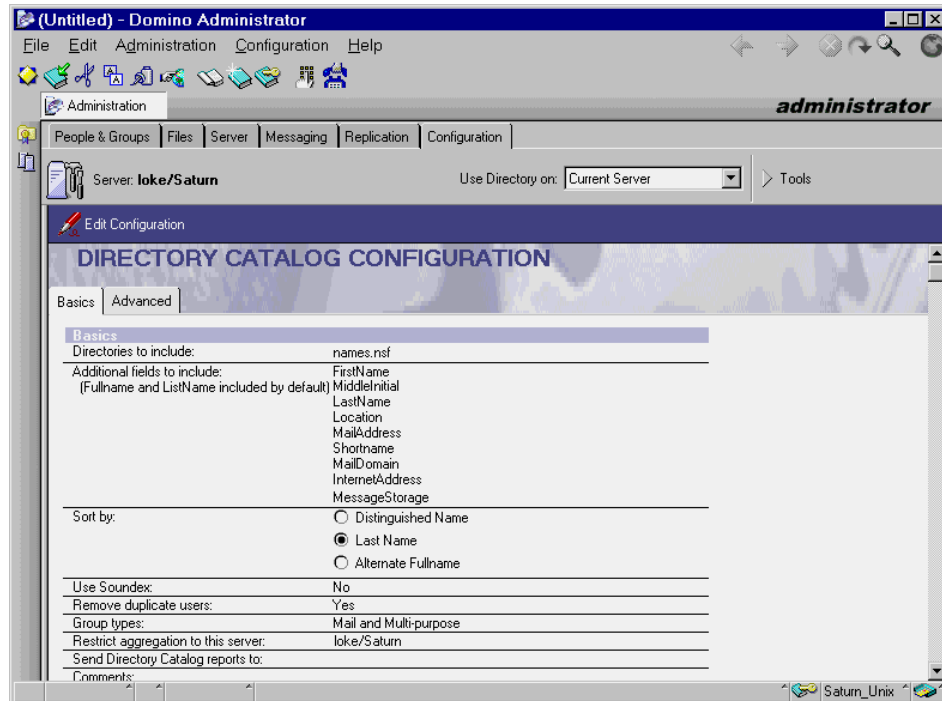


Figure 8-1 Creating a new Directory Catalog configuration document.

5. On the Basics tab, in the “Directories to include” field, list the filenames of the Domino Directories to be included in the Directory Catalog, for example, names1.nsf, names2.nsf, names3.nsf, names4.nsf.
6. Enter the name of the Domino server that runs the Directory Catalog Task in “Restrict aggregation to this server” to ensure that the Dircat task runs only on one server.
7. Close and save the document.

For more information, see “Creating a Source Directory Catalog” in Chapter 6 in *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341.

Running the Dircat task on schedule

Run the Dircat task to initially populate the source Directory Catalog and later to keep the entries in the source Directory Catalog synchronized with corresponding entries in the full Domino Directories. Use the following steps to run the Dircat task:

1. Open the Server document in edit mode for the server that stores the source Directory Catalog.

2. Click the Server Tasks - Directory Cataloger tab.

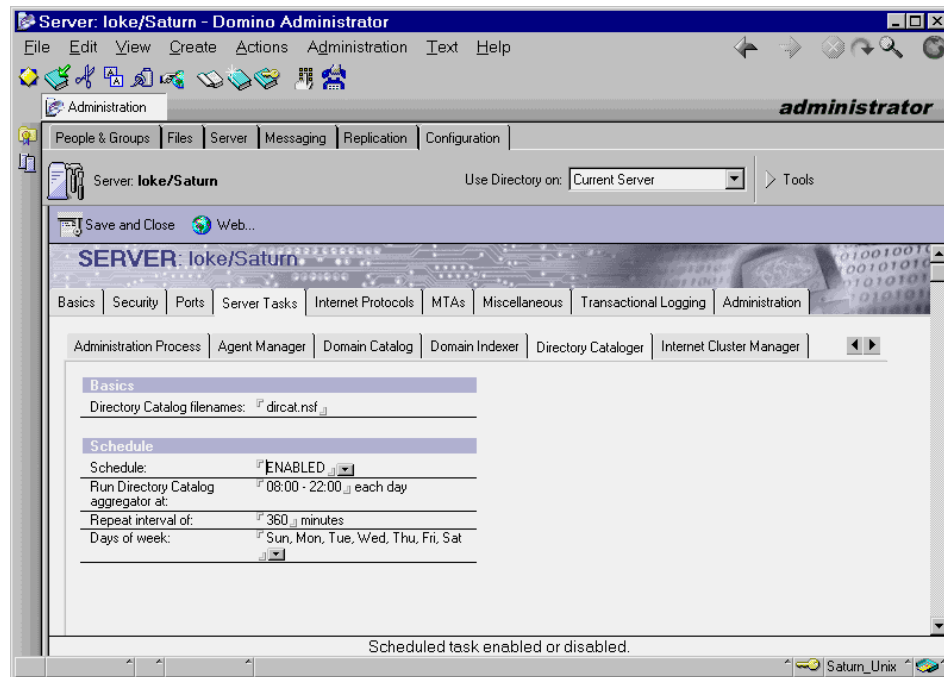


Figure 8-2 Directory Cataloger schedule

3. Make entries in the appropriate fields.

Note: Always run the Dircat task on the server that stores the source Directory Catalogs. If you run it on more than one server, replication conflicts occur. Use the configuration field “Restrict aggregation to this server” to ensure that the Dircat task runs only on one server.

For more information, see “Building and Updating a Source Directory Catalog” in Chapter 6 of *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341.

Setting up the Directory Catalog on a server

If you’ve configured and built a source Directory Catalog for server use, set up the Directory Catalog on the servers throughout your organization that will use it. In addition to setting up the Directory Catalog, we also recommend that you set up Directory Assistance.

Use the following steps to set up the Directory Catalog on the servers.

1. Make sure you have already run the Dircat task to build the source Directory Catalog.
2. Create a replica of the source Directory Catalog on other servers in the domain that will use the catalog. Use the same file name as the source Directory Catalog for each replica. Domino automatically creates a full-text index for each replica.
3. From the Domino Administrator, in the server pane on the left, select a server that is in the same domain as the source Directory Catalog. If you don't see the server pane, click the Servers icon.
4. Click the Files tab.
5. Select the Domino Directory, and then double-click to open it.

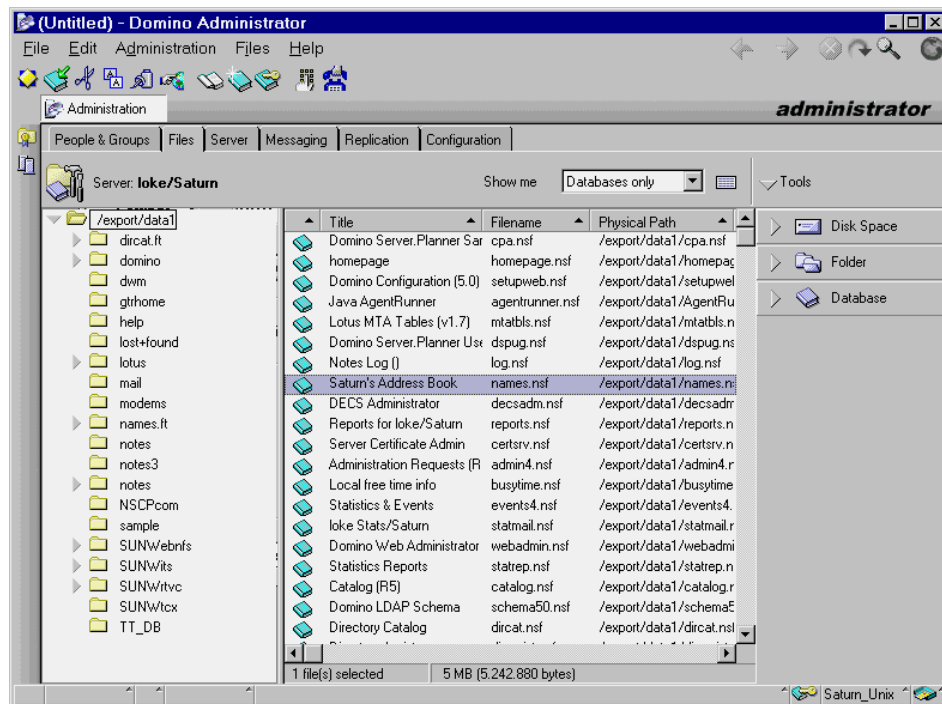


Figure 8-3 Editing the Directory Profile

6. Choose Actions - Edit Directory Profile. You will see the screen shown in Figure 8-4 on page 210.

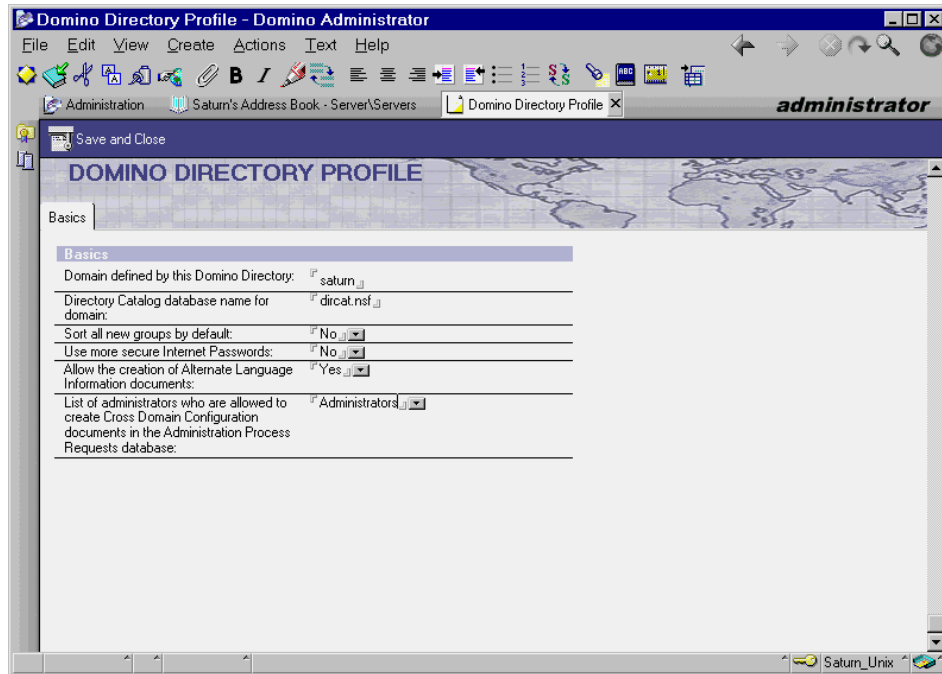


Figure 8-4 The Domino Directory Profile document

7. In the “Directory catalog database name for domain” field, enter the file name you chose for the Directory Catalog, and then click Save and Close.
8. Make sure there are connection documents that schedule replication between the server storing the source Directory Catalog and the servers on which you create replicas of the Directory Catalog. Scheduling replication ensures that replicas remain synchronized with the source Directory Catalog.
9. Wait for the file name to replicate to a particular server’s replica of the Domino Directory, or force the replication.
10. Restart the servers that have replicas of the Directory Catalog so the servers detect the file name. In Domino R5.0.2 a new command for restarting the Domino Server was introduced. Enter **Restart Server** from the remote server console of the Domino Administrator client. The Domino server stops and restarts after a short delay.

Setting up Mobile Directory Catalogs

If you’ve configured and built a source Directory Catalog for mobile use, you can create a User Setup Profile to set up the Mobile Directory Catalog on Notes clients. The User Setup Profile performs two tasks:

- ▶ It creates a replica stub (an empty replica) of the Mobile Directory Catalog. Replication between the Notes client and the Domino server populates the Mobile Directory Catalog and keeps it updated.
- ▶ It appends the Mobile Directory Catalog's file name to the contents of the "Local address books" field in the user preferences for mail.

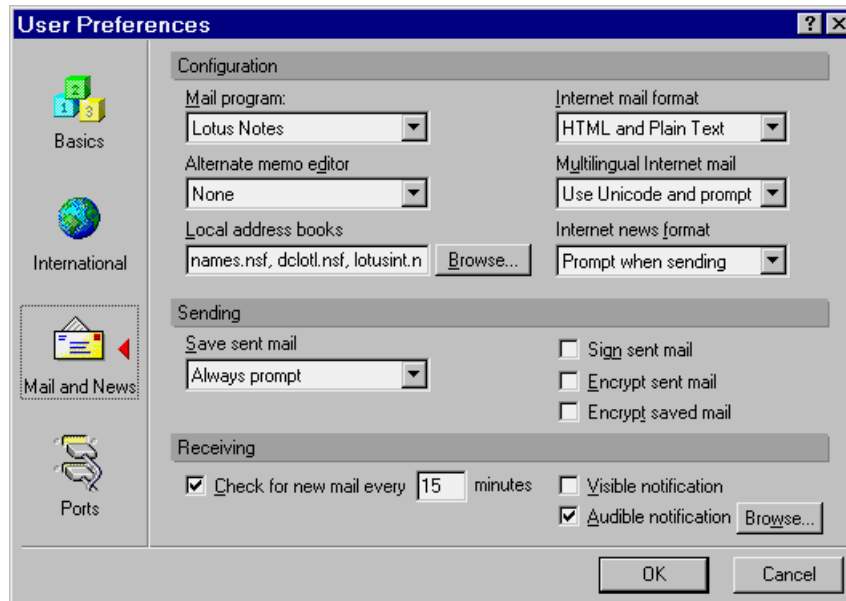


Figure 8-5 Verifying the Local address books setting in User Preferences

Note: If you don't use a User Setup Profile, each Notes user must manually perform these two procedures.

For more information, see "Setting up the mobile directory catalog" in the "Domino Directories" section of the Domino 5 Administration online help.

8.3 Directory Assistance

A new server gets its primary Domino Directory for its domain during the setup process. Each server stores its own primary Domino Directory under the file name names.nsf. Directory Assistance is a feature that enables users and servers to locate information in a directory that is not a server's primary Domino Directory.

You can set up Directory Assistance for:

- ▶ Secondary Domino Directories
- ▶ LDAP directories, including those on third-party servers

Include secondary Domino Directories in Directory Assistance to:

- ▶ Use the Domino Directories to authenticate Web clients that use the Domino Web service
- ▶ Allow Notes users to easily address mail to users registered in the directories
- ▶ Extend LDAP client searches to secondary Domino Directories

Include LDAP directories in Directory Assistance to:

- ▶ Use the directories to authenticate Web clients that use the Domino Web service
- ▶ Use one directory to verify Web clients' membership in groups in the directory
- ▶ Refer LDAP clients that connect to a Domino LDAP service to the directories
- ▶ Allow Notes users to use mail addresses of users in the LDAP directories

8.3.1 Setting up Directory Assistance

To set up Directory Assistance in a Domino domain, complete these steps:

1. Set up a Directory Assistance database.
2. Set up Directory Assistance for each secondary Domino Directory.
3. Set up Directory Assistance for each LDAP directory.

Each step is described in detail in the following sections.

Setting up a Directory Assistance database

1. Create a Directory Assistance database from the Directory Assistance template (da50.ntf).
2. Replicate the database to each server that needs it.
3. Identify the Directory Assistance database on servers that need it. Add the replica file name of the Directory Assistance database to the "Directory Assistance database name" field on the Basics tab in Server documents in the Domino Directory. You can manually enter the file name of the Directory Assistance database on one server document, or use the administration process to add the file name of the Directory Assistance database to multiple server documents. Figure 8-6 on page 213 shows the Server document after the Directory Assistance database has been added.

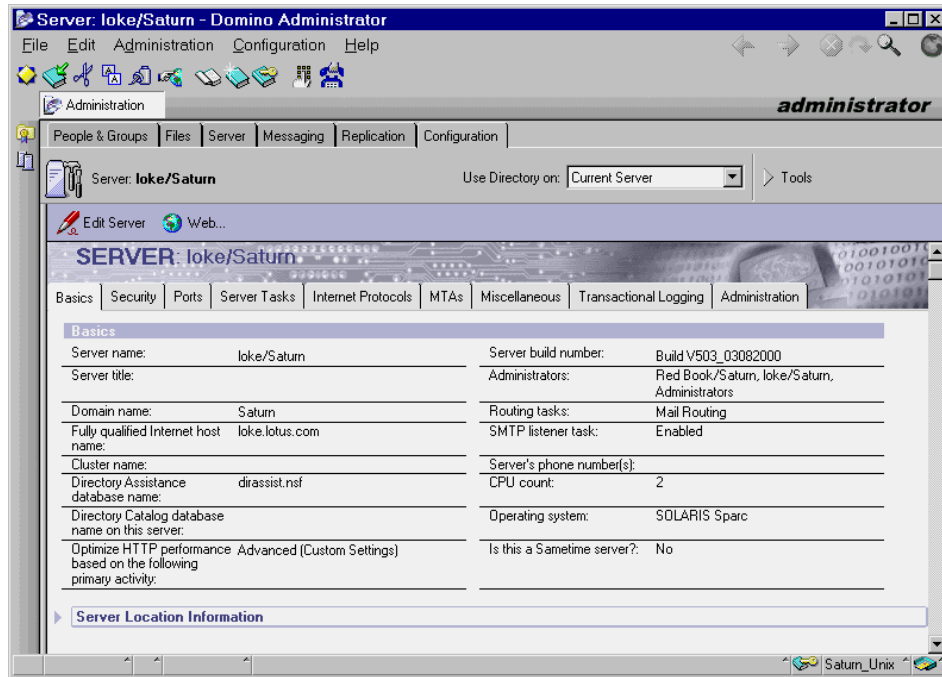


Figure 8-6 Setting up Directory Assistance

For more information, see “Setting up a Directory Assistance database” in the “Domino Directories” section of the Domino 5 Administration online help.

Setting up Directory Assistance for each secondary Domino Directory

After you set up the Directory Assistance database, follow these steps to set up Directory Assistance for a secondary Domino Directory.

1. Plan locations for replicas of the secondary directory.
2. Enable server and user access to the locations chosen for the secondary directory.
3. Create a Directory Assistance document for the secondary Domino Directory.

A Directory Assistance document is shown in Figure 8-7 on page 214.

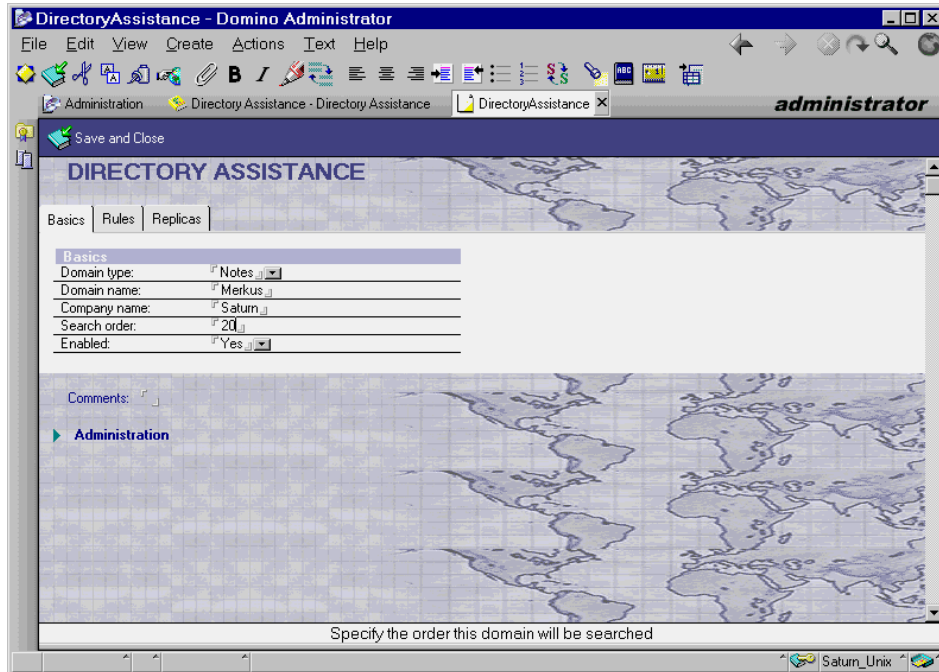


Figure 8-7 Directory Assistance document

Note: We recommend that you set up a Directory Catalog on servers that use Directory assistance.

For more information, see “Setting up directory assistance for secondary Domino directories” in the “Domino Directories” section of the Domino 5 Administration online help.

Setting up Directory Assistance for each LDAP directory

To set up Directory Assistance for an LDAP directory, you configure a single Directory Assistance document for the directory to do one or more of the following:

- ▶ Authenticate Web clients using credentials in an LDAP directory.
- ▶ Verify membership in a group that’s stored in an LDAP directory.
- ▶ Refer LDAP clients to an LDAP directory.
- ▶ Use an LDAP directory to verify mail addresses on behalf of Notes users.

Figure 8-8 shows an example of a Directory Assistance document for referring LDAP clients to an LDAP Directory.

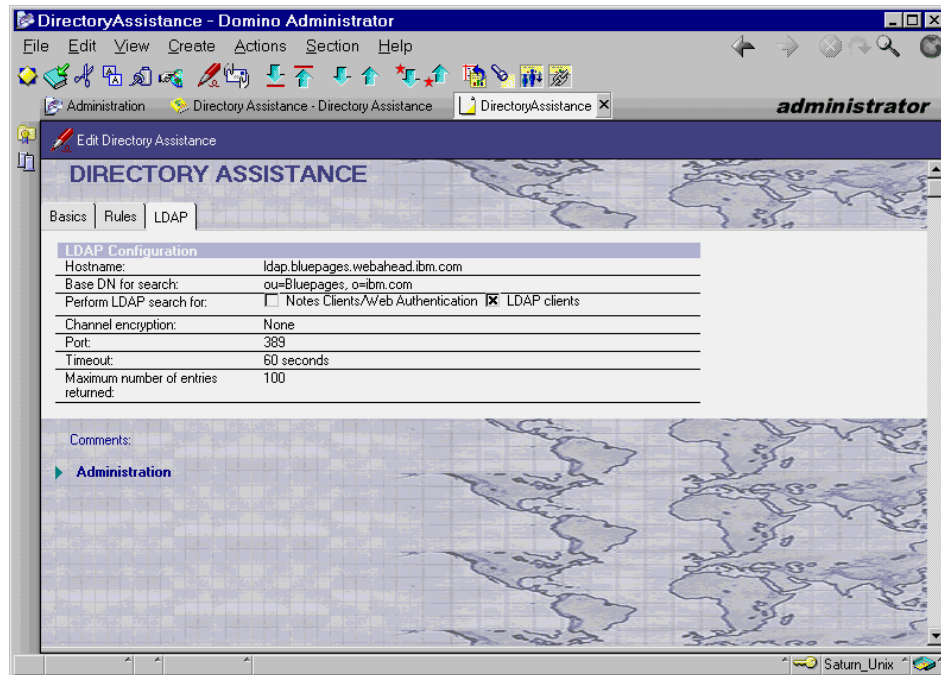


Figure 8-8 Configuring Directory Assistance for LDAP

For more information, see *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341, and “Setting up directory assistance for LDAP directories” in the “Domino Directories” section of the Domino 5 Administration online help.

8.4 Extended Directory Catalog

The Extended Directory Catalog is a new feature to Domino Release 5.0.5 and as stated in the release notes it combines advantages of the Domino Directory and the Directory Catalog by aggregating entries from multiple Domino directories into a single directory database. The Extended Directory Catalog is created by means of the same process as the Standard Directory Catalog, but instead of using the DIRCAT50.NTF, the Extended Directory Catalog uses the PUBNAMES.NTF (Domino Directory) template to create the target database and therefore it retains the full set of indexed views and other features of the Domino directory. This enables the enterprise to maintain a single consolidated server-based directory structure that responds rapidly to a variety of search patterns and can contribute to enhanced mail router performance.

This hybrid design based on the Domino Directory provides more flexibility and faster responses locating entries because a server can virtually always use views to quickly look up names. In contrast, to look up names in a standard Server Directory Catalog created from the DIRCAT50.NTF template, a server must do full-text searches—a slower lookup process than view lookups—when the name formats don't correspond to the "Sort by" configuration setting. Since the Extended Directory Catalog contains the views that are in a standard Domino Directory and combines multiple directories into one database, it can be quite large. Therefore, don't replicate the database to Notes clients and use as few replicas on servers as feasible.

Servers use Directory Assistance to determine the locations of an Extended Directory Catalog. One Directory Assistance document, and therefore one set of naming rules, applies to all the directories aggregated into an Extended Directory Catalog.

8.4.1 How to setup Extended Directory Catalog

1. If you currently use the standard Server Directory Catalog, disable it by removing its file name from the "Directory Catalog database name on this server" field in the Basics tab of the Server documents. If you've specified the file name there rather than in Server documents, remove its file name from the "Directory catalog database name for domain" field in the Public Directory Profile document.
2. On the server that runs the Dircat task, use the File -> Database -> New command to create the Extended Directory Catalog from the PUBNAMES.NTF template. Give the database a unique file name and title; don't give it the file name NAMES.NSF. In our lab we chose the name extended.nsf.
Note: It is not necessary to create a full text index.
3. In the ACL of the database you created in step 2, set the Default access to "Reader."
4. Open the database you created in step 2, then choose Create -> Aggregate Configuration, fill out the Configuration document, and click Save and Close. This document has most of the same configuration choices as the Configuration document used in the standard Server Directory Catalog. However, if you want to include Server documents in the Extended Directory Catalog, you can do so by selecting the "Include Servers" option. Also, there is no "Sort by" option—the Extended Directory Catalog retains all the indexed views in the Directory, so this option is unnecessary. The Server - Aggregate Directory Configuration view shows the saved configuration document. Keep these points in mind when you configure an Extended Directory Catalog:

- Don't aggregate the primary Domino Directory into an Extended Directory Catalog.
- If the "Additional fields to include" configuration field is blank, the Dircat task aggregates all fields from the source directory documents. To use the Extended Directory Catalog for Web user authentication, you must use the "Additional fields to include" configuration field to aggregate additional fields. To use names and passwords to authenticate Web users, add the HTTP password field to the configuration. To use X.509 client certificates to authenticate Web users, add the UserCertificate field. Figure 8-9 shows an example configuration we used in our lab.

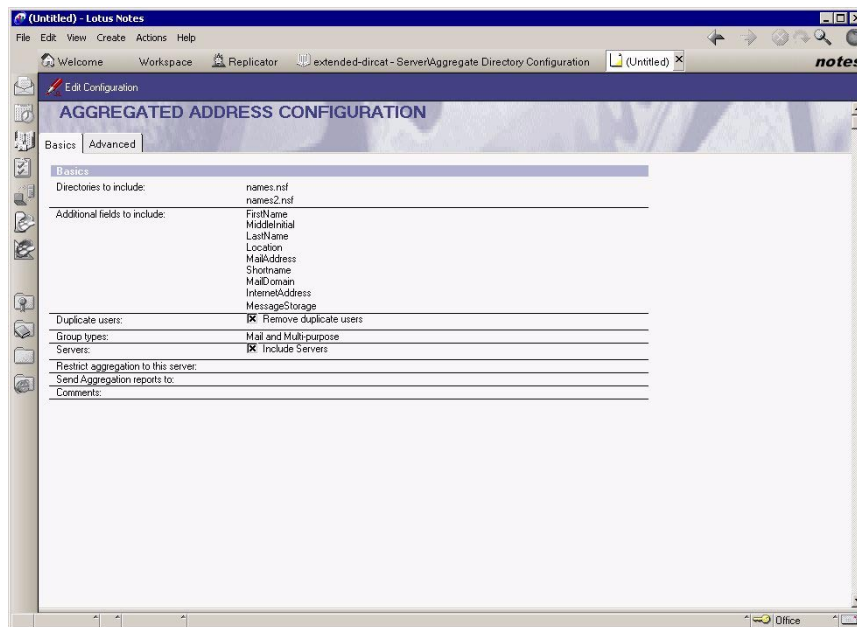


Figure 8-9 Aggregated Address Configuration - basic settings

5. To build the Extended Directory Catalog, run the Dircat task against the database you created. In our example, this was **load dircat extended.nsf**. Given the larger size of the Extended Directory Catalog, expect the Dircat task to take longer to run against an Extended Directory Catalog than it does on a Standard Directory Catalog. You can improve Dircat performance by selecting "No" next to the "Remove duplicate users" option. If you select "No," then entries with identical names are all included in the Directory Catalog and users choose between the duplicates. Selecting "No" avoids the building of a particular view used to ensure the removal of entries with duplicate names.
6. If you use Directory Assistance, open the Directory Assistance database and remove the Directory Assistance documents for all directories that you

included in the Extended Directory Catalog. If you don't currently use a Directory Assistance database, create one from the DA50.NTF template, replicate it to servers, and add its file name to the field "Directory Assistance database name" in the Basics tab of Server documents.

7. In the Directory Assistance database, create a Directory Assistance document for the Extended Directory Catalog. Choose Add Directory Assistance, fill out the configuration fields, then click Save and Close. Keep the following points in mind:
 - Next to Domain type, select Notes, not LDAP. Next to Domain name, make up a unique domain name.
 - Do not specify the name of the primary domain. If you want to trust the directory catalog for Web user authentication, include a rule that is "Trusted for Credentials."
 - In the replicas tab, specify one or more replicas of the Extended Directory Catalog. In a large domain it's important that there be more than one replica for performance and failover reasons.
8. Replicate the updated Directory Assistance database to the servers in the domain that will use it. Then restart the servers to load the new Directory Assistance information or wait 5 minutes for the servers to do this themselves.

8.5 Domino LDAP service

The Lightweight Directory Access Protocol (LDAP) is an open industry standard. LDAP defines a standard method for accessing and updating information in a directory. LDAP is gaining wide acceptance as the directory access method of the Internet and is therefore also becoming strategic within corporate intranets.

8.5.1 What is Domino LDAP service

Domino R5 includes a wide range of LDAP features, including support for LDAP V3.

Lotus Domino R5 includes two types of support for LDAP:

1. You enable the LDAP Service on a Domino server by starting the LDAP task on it. Users can execute directory operations, such as searching or modifying Domino Directory entries using an LDAP client.
2. Domino supports several LDAP features that you can use with a third-party LDAP directory server. The LDAP Service isn't required to use these features.

Lotus Domino R5 LDAP Service features

Domino R5 supports the following service features of LDAP V2 and V3:

- ▶ Different access protections, including anonymous access to specified fields, user and password authentication, SSL and X.509 certificates, and others.
- ▶ Support of third-party LDAP clients.
LDAP support in Domino makes directory information highly accessible. It enables any LDAP client, whether it's a POP3, IMAP, a browser, or a common mail client, to use the Domino Directory to look up names and addresses.
- ▶ Add, delete, and modify directory entries.
By default, LDAP write access to the Domino Directory is not allowed. You can enable LDAP write access to the Domino Directory by editing the Directory settings.
- ▶ Schema.
The schema for an LDAP directory defines how information is stored as entries in a directory. The smallest piece of information in a schema is an attribute. Attributes correlate to Domino Directory fields. Groups of related attributes are known as object classes. Object classes correlate to Domino Directory forms and subforms. The default Domino LDAP service schema includes many standard LDAP attributes and object classes, as well as some that are specific to Domino.
- ▶ Searches based on alternate languages.
You can create Alternate Language Information documents that allow LDAP users to search Person entries and retrieve the results using their native languages.

For more information on LDAP features in Domino, see “The Domino LDAP service” in the “Domino Directories” section of the Domino 5 Administration online help.

For more information on LDAP, see the redbook *Understanding LDAP*, SG24-4986.

8.5.2 Setting up Domino LDAP service

The Lotus Domino R5 LDAP server task can be installed automatically when you install Lotus Domino R5 or it can be added at any time afterwards. There is no need to install additional software to enable Lotus Domino to function as an LDAP server. There is also very little extra configuration needed.

To turn Lotus Domino into an LDAP server, set up the Domino server and set up security for the server, then use the following steps to implement the Domino LDAP service.

1. Create a full-text index for the replica of the Domino Directory on the server that runs the LDAP service.
2. Start the Domino server and the LDAP task.
3. If your organization uses more than one Global Domain document, you must specify the one that the LDAP service uses to return users' Internet addresses to LDAP clients. Open the Global Domain document. In the "Use as default Global Domain" field, choose Yes. A Global Domain document is used to specify the settings for all used LDAP directories.
4. Set up LDAP clients to connect to the LDAP service.
To use the Domino LDAP service, each LDAP user, whether Notes or non-Notes, must set up the client to connect to the LDAP service. For more information, see "Setting up users to use the LDAP service" in the "Domino Directories" section of the Domino 5 Administration on-line help.
5. (Optional) Customize the default LDAP service configuration. In most cases, the LDAP service functions correctly when using the default settings.
6. To check whether you set up the LDAP service correctly, use an LDAP client or the `ldapsearch` utility to issue a query to the LDAP service. Use of the `ldapsearch` utility is described later in this chapter.
7. To allow clients to connect to the LDAP service over the Internet, connect the server that runs the LDAP service to an Internet service provider (ISP) and register the server's DNS name and IP address with the ISP.

Note: TCP/IP port 389 and TCP/IP port 636 are the industry standard ports for LDAP connections over TCP/IP and SSL, respectively. You should use the default port numbers in most cases. Firewalls must pass traffic on these ports.

8.5.3 Starting and stopping the LDAP server task

There are three options for starting the LDAP server:

1. If you selected LDAP when you installed Lotus Domino, the LDAP server task starts automatically.
2. If you did not select LDAP when you installed Lotus Domino, you can start it manually using the following command at the server console:
load ldap
3. To start the LDAP server task automatically at Lotus Domino Server startup, add LDAP to the `ServerTasks` entry in `Notes.ini`. For example:
ServerTasks=LDAP, REPLICa, ROUTER, UPDATE, ...

You can verify that the LDAP server is running by using the command **show tasks** at the server console.

Figure 8-10 shows the result of the **show tasks** command.

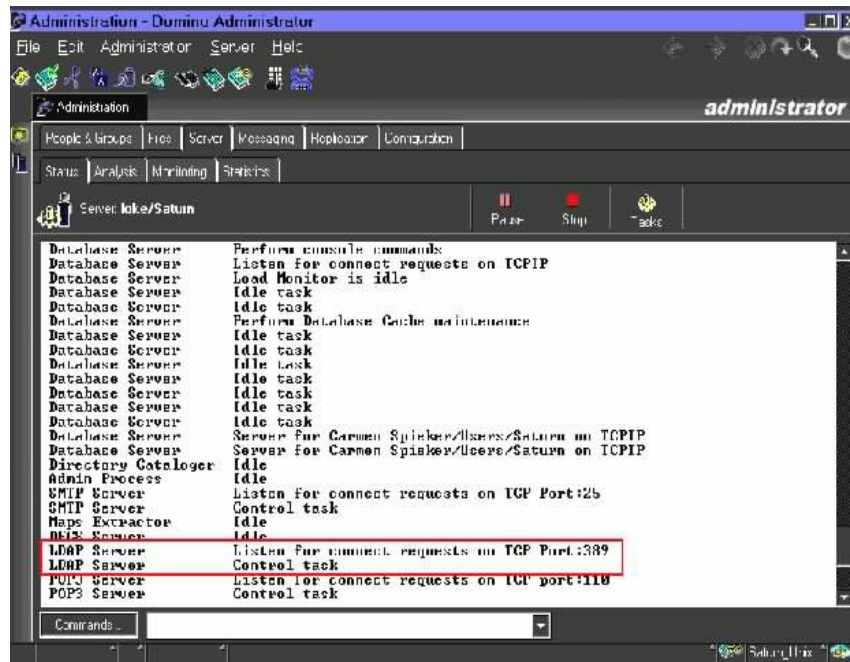


Figure 8-10 Verifying the LDAP server is running from the server console

To shut down the LDAP server, you have several options:

1. Shut it down manually by executing the command **te11 ldap quit** at the server console.
2. Shut down the entire Domino server.
3. Deactivate the automatic startup by removing LDAP from the ServerTasks entry in Notes.ini and then restarting the Lotus Domino server.

The **show tasks** console command should no longer show LDAP entries.

8.5.4 Showing LDAP statistics

There are several ways to display LDAP statistics:

- ▶ On a remote console
- ▶ On a browser, with the Web Administrator
- ▶ On the Server - Status tab

To view LDAP statistics on a remote console, perform the following steps:

1. Open the Domino Administrator client.
2. Select Server from the server list.
3. Click the Server - Status tab.
4. Click the Console button to open the remote console.
5. Enter the command **sh stat ldap** in the command line and press Enter.
6. The screen shown in Figure 8-11 is displayed.

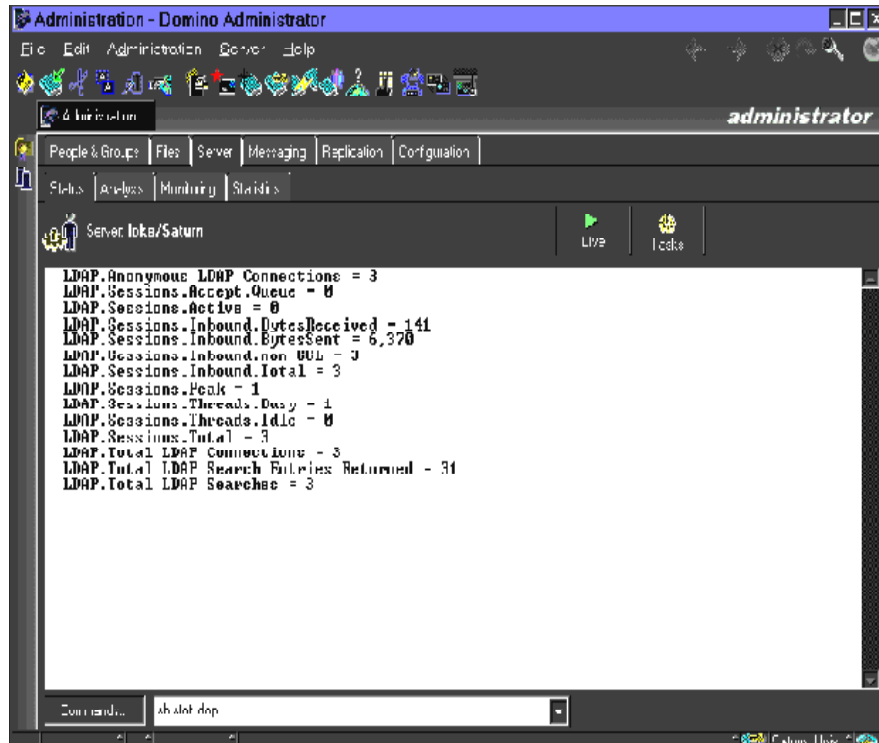


Figure 8-11 Viewing LDAP statistics from the server console.

To view LDAP statistics with the Web Administrator, use the following steps:

1. Start the Web browser.
2. Connect to the Domino server's Web Administrator (<http://dominoserver/webadmin.nsf>).
3. Use the console command system to **sh stat ldap**

To view LDAP statistics on the Server - Status tab, use the following steps:

1. Open the Domino Administrator client.

2. Select Server from the server list.
3. Click the Server - Status tab.
4. Click the LDAP twisty to view LDAP statistics.
5. The screen shown in Figure 8-12 appears.

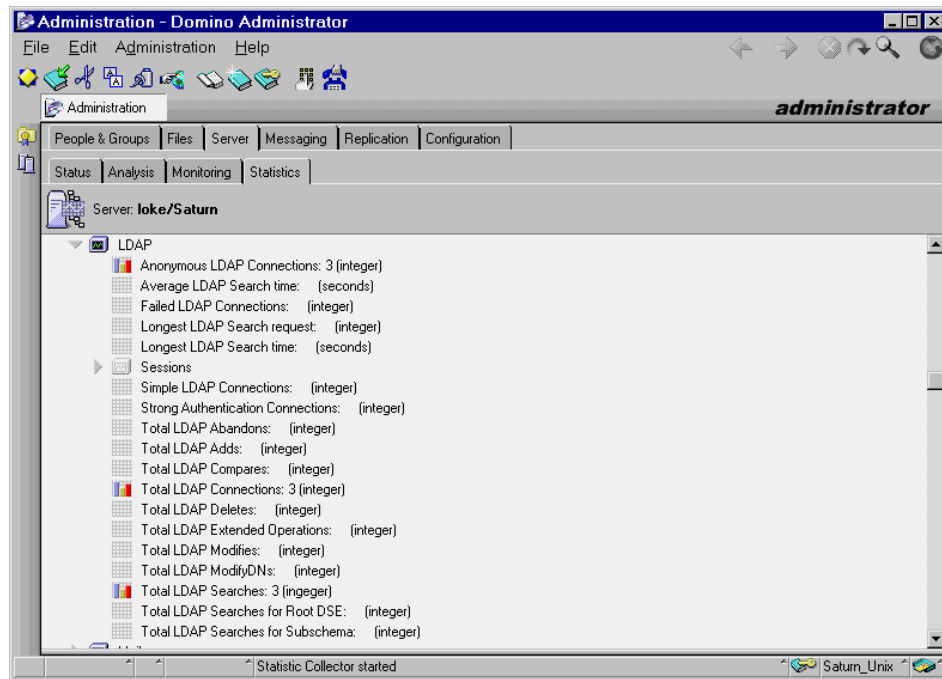


Figure 8-12 Viewing LDAP statistics using the Administration client.

8.5.5 Using the `ldapsearch` utility to search LDAP directories

Domino provides a command-line search utility, `ldapsearch.exe`, that allows you to use LDAP to search entries in the Domino Directory on a server that runs the LDAP service, or to search entries in a third-party LDAP directory. Note that you do not have to enter the command on a machine that runs the Domino LDAP service. The `ldapsearch` utility connects to the server that you specify and returns results according to the search criteria. It is available on Domino server and Notes client platforms.

Note: To use this tool, the `Notes.ini` file must be included in your path statement.

Performing a search with the ldapsearch utility

Enter the following command:

```
ldapsearch parameters searchfilter attributes
```

where:

- ▶ *parameters* are case-sensitive command-line parameters. For more information, see “Using parameters with ldapsearch” in the “Domino Directories” section of Domino 5 Administration online help.
- ▶ *searchfilter* is a required search filter that causes ldapsearch to find only entries that meet specific attribute criteria.
- ▶ *attributes* are options that limit the values that ldapsearch returns. Separate each attribute with a space. If you don’t specify one or more attributes, ldapsearch returns all attributes.

Example 8-1 shows the result of the ldapsearch command:

```
ldapsearch -h saturn.lotus.com objectClass=*
```

The search connects to the LDAP service on host saturn.lotus.com and returns all attributes and values.

Example 8-1 LDAP search results

```
$ ldapsearch -h saturn.lotus.com "objectClass=*"
CN=Administrators
cn=Administrators
mail=Administrators@lotus.com
objectclass=top
objectclass=groupOfName
objectclass=dominoGroup
member=CN=Red Book,OU=Users,O=Saturn
member=CN=David Morrison,OU=Users,O=Saturn
member=CN=Carmen Spieker,OU=Users,O=Saturn
member=CN=Valerie Walker,OU=Users,O=Saturn
member=CN=Eric Dolce,OU=Users,O=Saturn
member=CN=Chris Odonnell,CN=,OU=Users,O=Saturn

CN=LocalDomainServers
cn=LocalDomainServers
mail=LocalDomainServer@lotus.com
objectclass=top
objectclass=groupOfName
objectclass=dominoGroup
member=CN=saturn,O=Saturn
member=CN=loke,O=Saturn
```

8.5.6 Exporting Domino Directory information

You can export Domino Directory information in a format that can be understood by other LDAP-compliant directories by extracting the Domino Directory information into a text file. The extraction leaves the data in Lightweight Data Interchange Format (LDIF), the RFC-compliant format used by LDAP servers and clients. LDIF defines a universally understood format used by LDAP servers to build their respective schema.

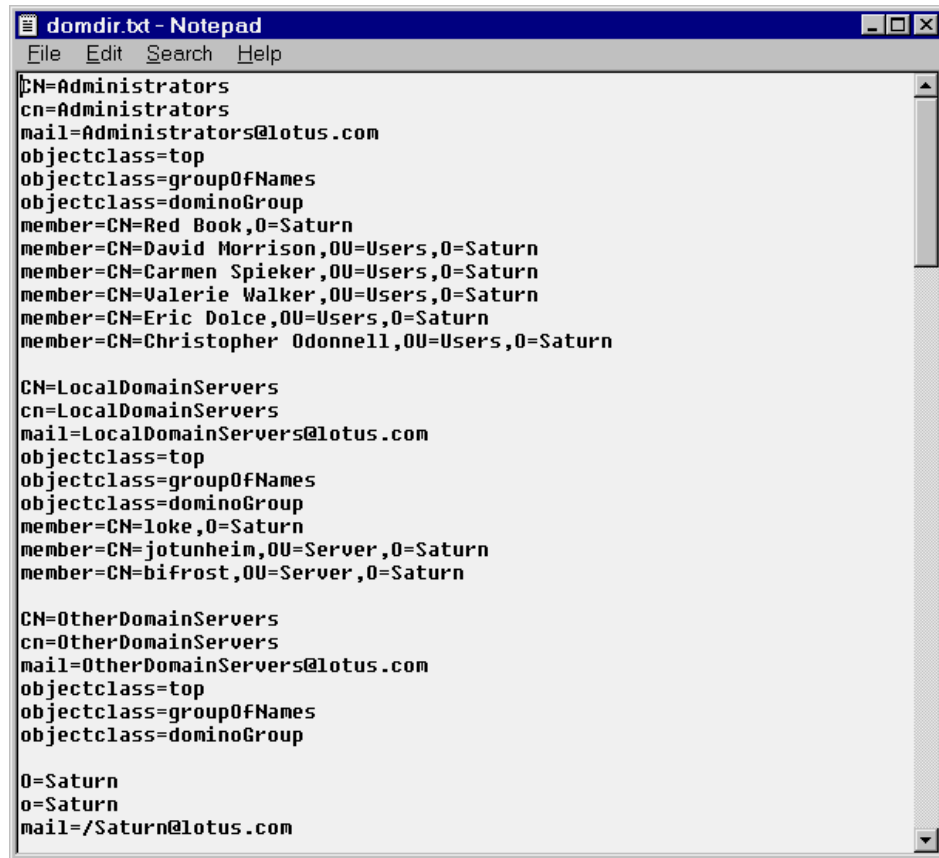
Use the following `ldapsearch` command to extract the Domino Directory information to a text file.

```
ldapsearch -h LDAPservername objectclass=* >filename.txt
```

For example:

```
ldapsearch -h saturn.lotus.com objectclass=* >domdir.txt
```

The text file created, shown in Figure 8-13 on page 226, can then be imported to another LDAP server.



```
domdir.txt - Notepad
File Edit Search Help

CN=Administrators
cn=Administrators
mail=Administrators@lotus.com
objectclass=top
objectclass=groupOfNames
objectclass=dominoGroup
member=CN=Red Book,O=Saturn
member=CN=David Morrison,OU=Users,O=Saturn
member=CN=Carmen Spieker,OU=Users,O=Saturn
member=CN=Valerie Walker,OU=Users,O=Saturn
member=CN=Eric Dolce,OU=Users,O=Saturn
member=CN=Christopher Odonnell,OU=Users,O=Saturn

CN=LocalDomainServers
cn=LocalDomainServers
mail=LocalDomainServers@lotus.com
objectclass=top
objectclass=groupOfNames
objectclass=dominoGroup
member=CN=loke,O=Saturn
member=CN=jotunheim,OU=Server,O=Saturn
member=CN=bifrost,OU=Server,O=Saturn

CN=OtherDomainServers
cn=OtherDomainServers
mail=OtherDomainServers@lotus.com
objectclass=top
objectclass=groupOfNames
objectclass=dominoGroup

O=Saturn
o=Saturn
mail=/Saturn@lotus.com
```

Figure 8-13 Contents of the text file

The text file created can then be imported to another LDAP server.

8.6 Summary

In this chapter we have given an overview of Domino Directory services.

We described the primary Domino Directory (names.nsf) with its documents, how to set up a directory catalog, directory assistance, extended directory catalog, and the Domino LDAP service. Finally, we explained how to use the `ldapsearch` utility and to export Domino Directory information.



Domino R5 as a Web server

In this chapter we describe how to configure a Domino server to work as a Web server.

We discuss in detail a number of changes that were implemented in the HTTP task for Domino R5, improving the performance and scalability over previous releases. One of the most significant is the possibility of having a cluster environment for the HTTP processes using the Internet Cluster Manager (ICM) process.

9.1 Solaris Operating System configuration

The considerations introduced in Chapter 4, “Tuning Domino Server on Solaris” on page 91 are still valid for the Domino Web server.

Considering the temporary nature of connections under the HTTP protocol (each request opens a connection, sends the message, returns the response, and closes the connection), particular care must be taken in configuring the TCPIP part of the Solaris Operating System.

9.1.1 Basic recommendation

It is possible that some other HTTP server could be running on your system, like Netscape or Apache. The only precaution is to check if other HTTP daemons are running on the Solaris system using the default port 80.

Use the **ps -ef** command and pipe the output to the **grep** command to check this:

```
# ps -ef | grep http
```

Note: The UNIX **grep** command searches a file for a pattern. It also reads from the standard input so it can be used in a pipeline command.

You should not see any HTTP-related task running on your system.

Use the **netstat** command to see if any daemons are using port 80:

```
# netstat -a | grep "\.80"
```

Note: The backslash (\) before the dot in the "\.80" string is used to avoid the meaning of the dot in a regular expression. This is a common practice if you want to use characters literally and avoid their specialized meaning in a UNIX command.

In this case the command should not have any output. If there are some daemons listening on port 80 you may have output like this:

```
*.80 *.* 0 0 0 0 LISTEN
```

Generally you can have other HTTP processes running on your system, listening on different ports. Running other HTTP systems on the same Solaris machine is not recommended if you want to have a high performance Domino Web server.

9.1.2 Network tuning

Some TCP parameters can be tuned to increase the Domino Web server performance.

CLOSE WAIT state

When a TCP connection is closed by the Domino Web server, it remembers the connection for a few minutes to make sure it can identify any leftover packets in transit over the network from clients related to that connection. The TCP standard defines this interval as twice the maximum life of a TCP packet, 120 seconds. The default value for Solaris is 240 seconds.

When high connection rates occur, this mechanism may not work as expected. A symptom of this is when the **netstat** command shows a lot of TCP connections in the CLOSE_WAIT status. To view this, type **netstat -a** on the command line.

One way to work around this problem is to increase the number of buckets in an internal TCP hash table. To do this you have to change the `tcp_conn_hash_size` kernel parameter using the **ndd** command. If you want to increase the size to 256 K entries the command is:

```
# /usr/sbin/ndd -set /dev/tcp tcp_conn_hash_size 262144
```

The command is accepted by the system immediately. To retain the change after an OS reboot place the following line in the file `/etc/system`:

```
set tcp:tcp_conn_hash_size=262144
```

TCP internal buffer

To increase the internal buffering of the TCP stack to deal efficiently with “bursty” Web traffic, edit the `/etc/system` file, adding the following entry:

```
set sq_max_size=512
```

Some tests made by Sun engineers have indicated that this feature helps Domino Web performance.

Note: This Solaris kernel parameter is not documented or supported by Sun, so it may change in the future.

TCP backlog settings

On heavily used Web servers, you will want to resize several queues that can fill up with waiting connections. The sample settings given below were for an 8 CPU machine; you may want to adjust these based on your own configuration.

To increase the maximum number of completed connections waiting to return from an accept call, add the following line to the `/etc/system` file:

```
set tcp_conn_req_max_q=4000
```

The following entry in `/etc/system` increases the maximum number of connections with the handshake incomplete:

```
set tcp_conn_req_max_q0=4000
```

These settings set the maximum limits within the Solaris networking code. To specify a higher backlog value while making a `listen()` call within Domino, add the following line to the `httpd.cnf` file in the Domino data directory:

```
listenbacklog      4000
```

9.2 Domino Web server configuration

The configuration of the HTTP server in Domino R5 is a very easy task. Most of the work is done at Domino installation time if you check the options to install the HTTP task.

If you choose to install the HTTP task after the setup process, you will find the HTTP name in the `Notes.ini` file to the `ServerTasks` entry:

```
ServerTasks=replica,router,update,amgr,adminp,HTTP
```

Tip: The content of the `Notes.ini` file is *not* case sensitive, so there is no problem if the name of the task is written with capitals and the effective name of the binary file is `http`. Remember that UNIX *is* case sensitive.

9.2.1 Settings on a Domino Web server

To change the settings of the Domino Web server, use the following steps:

1. Start the Domino Administrator.
2. Choose the server you want to reconfigure.
3. Choose the Configuration tab.
4. Choose server - All Server Documents.
5. Double-click the Domino server you want to change or select the server and click Edit Server.

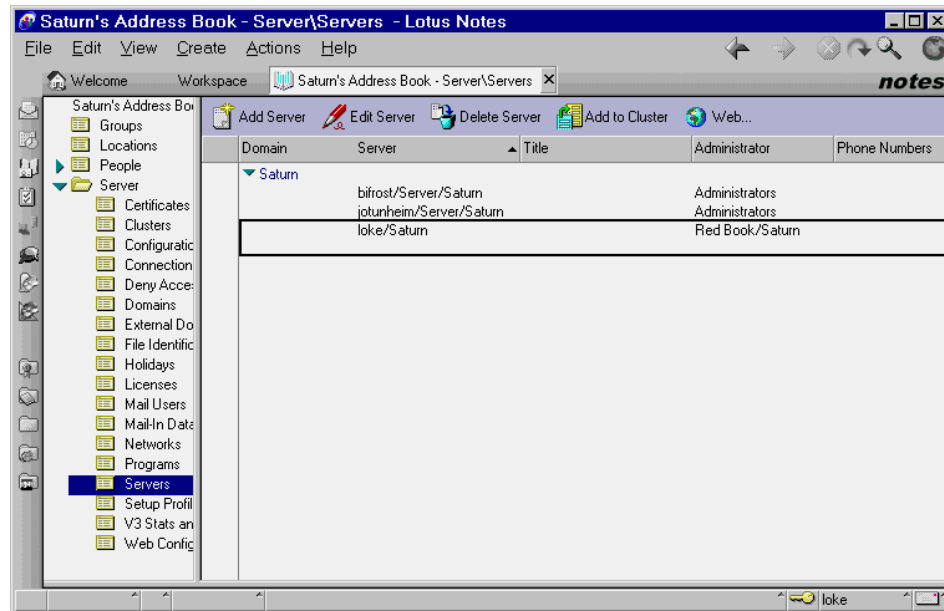


Figure 9-1 Web server configuration

To change the Domino Web server port, click Ports - Internet Ports in the server document. The Web tab should be selected by default.

It is best to use the default port 80 for a non-secure Web server and port 443 for a secure Web server.

Note: The secure server will not run until you create a server certificate. See the section “Setting up SSL on a Domino server” in the Domino 5 Administration online help.

Here you can also choose if you want to allow name and password authentication for clients connecting over TCP/IP; the default is “Yes.” Also specify whether you will allow anonymous connection over TCP/IP; again, the default is “Yes.” The same is true for the SSL protocol.

Next, select Internet Protocols - HTTP. In this section, you should make at least the following changes:

- In the Basic section select the “Bind to host name” option if you use a Domino partitioned environment. The parameters “Maximum request over a single connection” and “Number of active threads,” which are discussed later in this chapter, should be set. See 9.4, “Performance” on page 241 for more information.

- ▶ Enable logging, either to log files or to Domlog.nsf, if you want to create statistics telling you about, for example, who, how much, and which pages were accessed on your Web server. See 9.10, “Domino log and analysis tools” on page 266 for details.
- ▶ In the Mapping section, customize the Home URL. It should be either a Notes database or an HTML file.

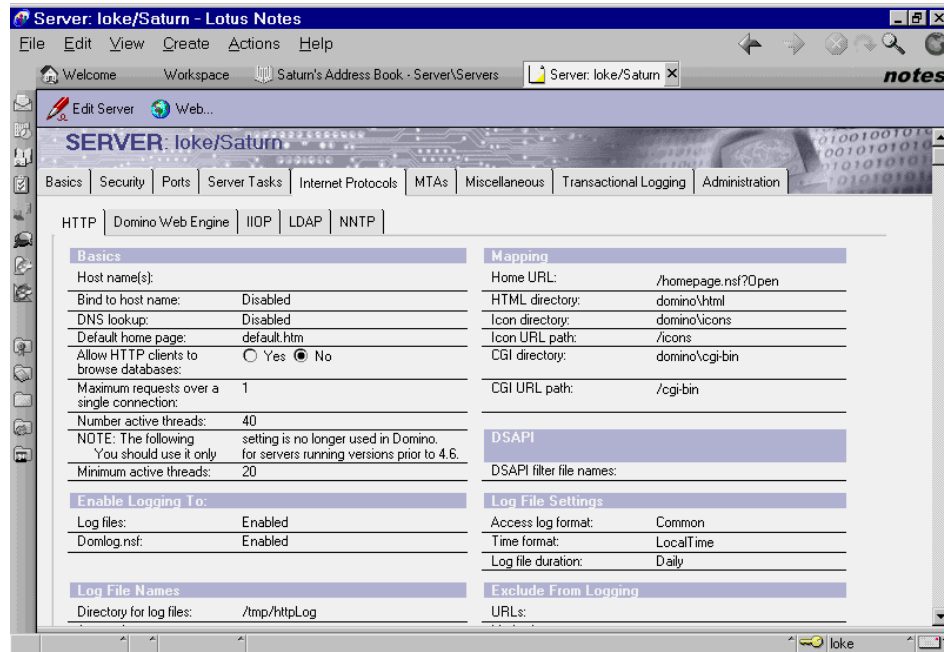


Figure 9-2 HTTP server setup

DSAPI section

The Domino Security Application Programming Interface (DSAPI) is a C API for writing your own extensions to the Domino Web server. A DSAPI extension, or “filter,” is a program you create that is notified when certain events occur in the Web server, such as when a URL request is received or when a user is about to be authenticated. Currently, DSAPI is used to create your own authentication programs and capture Web server events that can be used in billing applications.

In the field “DSAPI filter file names,” enter the shared object library names in the order you want them to be called.

Tip: The shared object should be built as libxxx.so, and referred to as xxx.so in this field.

9.2.2 Starting, stopping, and refreshing the Domino Web server

There are two ways to start the Domino Web server:

- ▶ Manually, by entering **load http** at the server console
- ▶ Automatically at Lotus Domino R5 start-up, by adding it to the ServerTasks in Notes.ini

You can start only one HTTP task per Domino server; you have to use the Domino partitions feature to have more than one HTTP task running in the Solaris box.

To stop the Web server, enter the command **tell http quit** at the server console, or remove HTTP from the ServerTasks in Notes.ini to stop it from starting at the next restart of the Domino server.

Type the command **tell http restart** at the server console to refresh the Web server, and if you made changes in the Domino Directory related to the HTTP configuration.

Tip: You can use the **server -c** Domino command to send a Domino console command from a UNIX prompt. Type **server -c "tell http quit"** to stop the HTTP task from a UNIX prompt.

9.3 Security

In this section we describe the new Web security features in Domino R5. Some new security features were added to Domino R5, such as Web Realm and File Protection.

9.3.1 Internet certificates

Domino certificate authorities can also issue Internet certificates to Notes users, Internet clients, and Internet servers. The Domino certificate authority issues signed X.509 format certificates that uniquely identify the requesting client or server. Internet certificates are required when sending encrypted or electronically signed S/MIME mail messages and when using SSL to authenticate a client or server.

S/MIME is a protocol used by clients to sign mail messages and send encrypted mail messages over the Internet to users of mail applications that also support the S/MIME protocol.

Domino R5 provides native X.509 V3 support along with the Notes certificate.

9.3.2 Browsing Domino databases via the Internet

A common security issue is accessing the log.nsf database via a Web browser, for example:

`http://www.acme.com/log.nsf`

Although the log.nsf database does not contain critical information, a Domino system that allows access to the system log is not secure.

To avoid this you have to change the ACL of the database to:

- ▶ Default No Access
- ▶ Anonymous No Access

You have to do this in each Domino database in your data directory that must be kept inaccessible to Internet users.

Tip: If you want to have manager access in the log.nsf database, you have to do this at installation time, when the setup asks you in which databases do you want to be manager.

9.3.3 Domino banner

When you go to a Domino site and you request a page, the HTML source has this header:

```
<HTML>
<!-- Lotus-Domino (Release 5.0.8 - 16 August 2001 on Solaris Sparc) -->
<HEAD>
```

This banner will be suppressed if you add the following line in your Notes.ini:

```
DominoNoBanner=1
```

This is a security precaution to hide the OS and Web server information from unauthorized users.

9.3.4 Session authentication

Session authentication is a new feature in Domino R5. A *session* is the time during which a Web client is actively logged on to a server. Session-based name-and-password security includes additional functionality that is not available with basic name-and-password security.

Session-based authentication creates a temporary cookie that stores the user name and password on the browser client. As the user traverses the site, responses for name and password are provided by the cookie.

This cookie passes the user credentials for every database within the Domino site, thus alleviating concerns of realm-based authentication.

Tip: If you wish to retain realm-specific logins, session-based authentication cannot be used.

Once a user logs in to the Web site, the credentials are passed to every database hosted by the server. The user login information, however, is not shared across virtual hosts or virtual servers; it is based on the host name of the URL request.

You can configure session authentication on the Domino Web Engine tab of the server document, in the HTTP Sessions section. You can configure the following two parameters:

- ▶ Idle session timeout: the server drops the user's credentials after the specified amount of inactive time.
- ▶ Maximum active sessions: the maximum number of user sessions allowed on the server at the same time.

These settings are shown in Figure 9-3 on page 236.

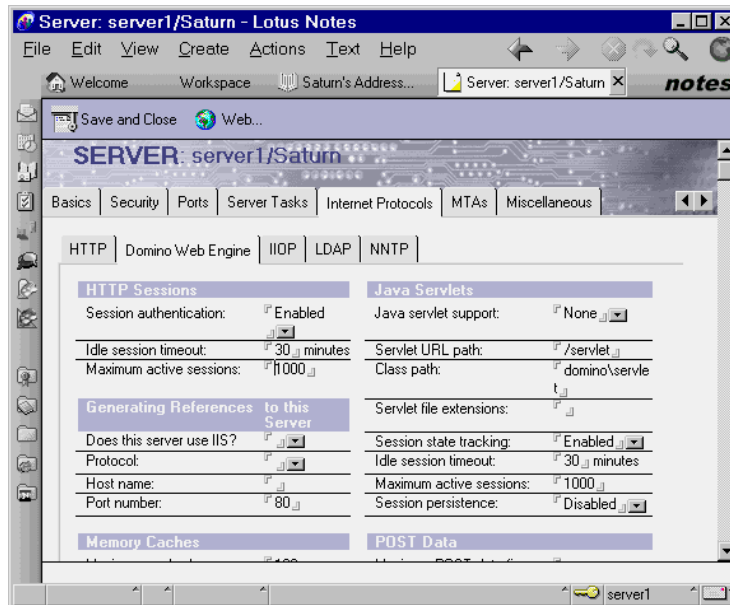


Figure 9-3 Session authentication settings in the server document

With the Session Authentication feature enabled, you can use the following command to find out who is using a Web browser to access your Domino 5 server:

```
> tell http show users
> 03/24/2000 11:00:07 AM There are 2 current HTTP user sessions
03/24/2000 11:00:07 AM User Name IP Address Expires
03/24/2000 11:00:07 AM red book 9.95.36.113 11:29:52 AM
03/24/2000 11:00:07 AM red book 9.95.36.113 11:29:28 AM
```

The session authentication feature is based on the cookie mechanism; it allows a Web server to store pieces of information on the client computer through the Web browser. These pieces of information, known as cookies, are stored on the client machine.

Tip: To return the value of a cookie, add a computed field called HTTP_COOKIE to your form using an empty string as a formula. This field will be populated with the cookie information. You can then use the field HTTP_COOKIE in other formulas on the page.

9.3.5 Domino Web Realms

To minimize the need for a Web user to repeatedly supply their password, Domino R5 administrators can set up Web Realms on the server. Realms, based on ACLs, are zones of file protection on a Web site.

The browser automatically stores and sends the credentials for pages in the same Realm, so the user can move throughout the Realm after supplying the password just once.

Access the page for setting up Realms by choosing Actions -> Web -> Create Realm. The resulting screen is shown in Figure 9-4.

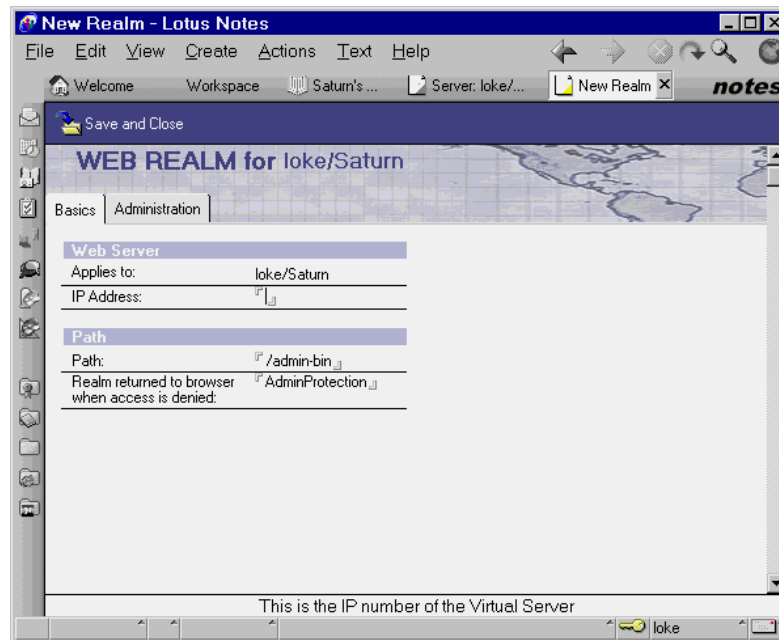


Figure 9-4 Web Realm: Basic setting

If you want users to be able to navigate freely throughout the directory OTSOREalm, located in the Domino data directory, and in the two subdirectories Sub1 and Sub2 (the hierarchy is shown in the Figure 9-5 on page 238), put the directory name OTSOREalm in the "Path" field in the Web Realm document.

Users with the right authorization will then be able to access all databases located in Sub1 or Sub2 without retyping their authentication.

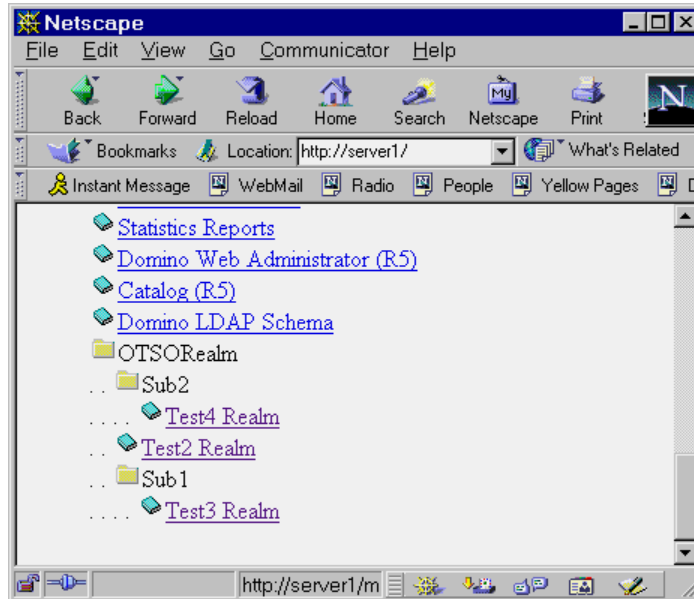


Figure 9-5 Displaying web realms in Netscape

9.3.6 Domino File Protection

In Domino R5, Protect Directives in the file `httpd.cnf` are replaced by File Protection documents stored in the Domino Directory database. Domino R5 now ignores any Protect Directives it finds in the `httpd.cnf` file.

File Protection documents control the access that Web browser clients have to the files. You can enforce file system security for files that browser users can access. For example, for HTML, JPEG, and GIF, you can specify the level of access for these types of files and the names of the users who can access them.

You can apply file system protection on CGI scripts, servlets, and agents. However, the file protection does not extend to other files accessed by the scripts, servlets, or agents. For example, you can apply file protection on a CGI script that restricts access to a group named "Web Admins." However, if the CGI script executes and opens other files (or causes other scripts to be executed), the File Protection document is not checked to determine whether "Web Admins" has access to these files.

File protection also does not extend to files in the following directories, which contain default image files and Java applets that are used by the HTTP web server and other applications (for example, mail databases):

- ▶ lotus/notes1/data/domino/java, accessed via Web browser using the path `http://server/domjava`
- ▶ lotus/notes1/data/domino/icons, accessed via Web browser using the path `http://server/icons`

File system protection does apply, however, to files that access other files, for example, HTML files that open image files. If a user has access to the HTML file but does not have access to the JPEG file that the HTML file uses, Domino does not display the JPEG file when the user opens the HTML file.

You have to consider setting up File Protection documents for each directory Web users are able to access. There is no file protection for an upgraded or new R5 server until you create File Protection documents.

You do this by choosing Actions -> Web -> Create File Protection. The resulting screen is shown in Figure 9-6.

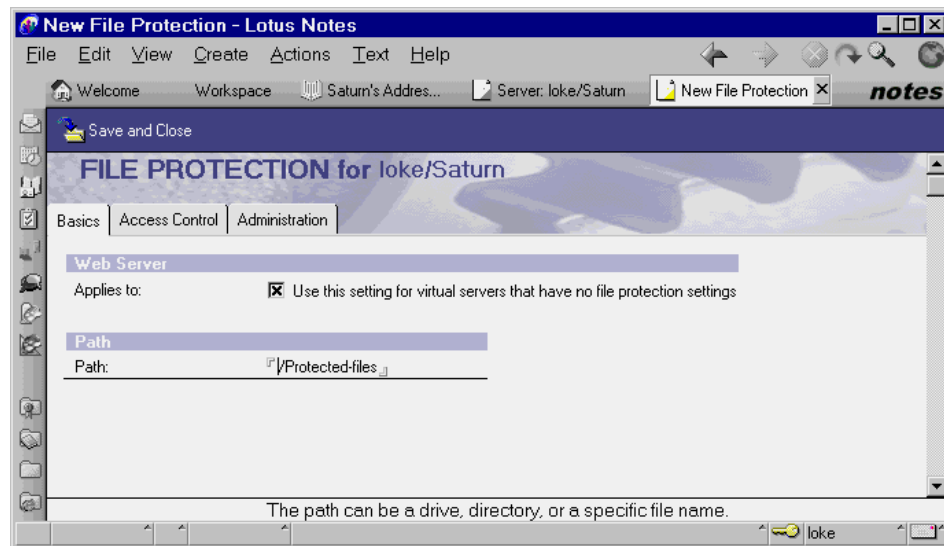


Figure 9-6 File Protection: Basic setting

The ability to set file protection might be needed in mixed environments, where you have some data in the Notes databases and other data in text files. These protection settings apply to all Web servers on a Lotus Domino R5 server.

You can only grant access to users specified in the server's Domino Directory, even if you are allowed to enter any user. You assign these permissions by clicking "Set/Modify Access Control List" in the Access Control tab.

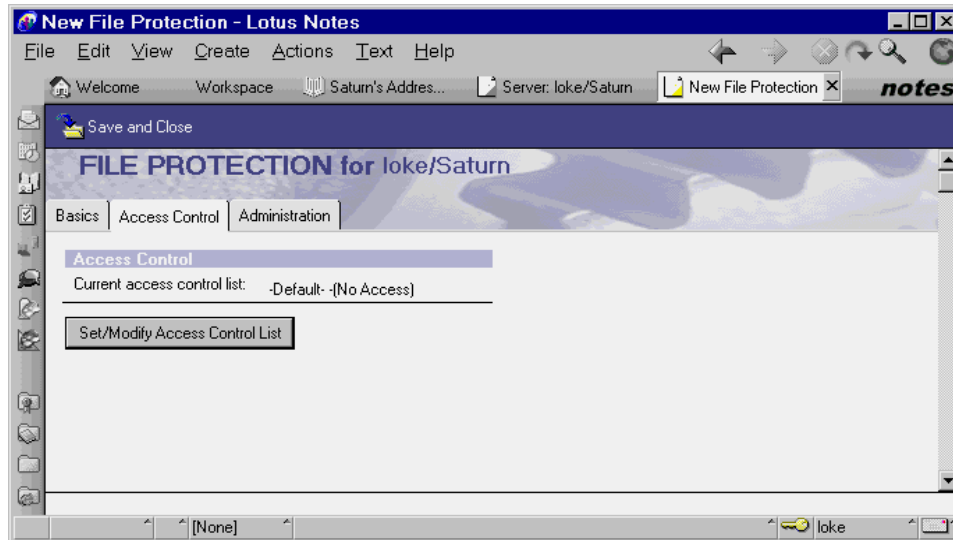


Figure 9-7 Access control for file permissions

There are three access levels you can assign to a user:

- ▶ Read/Execute access (GET method)
- ▶ Write/Read/Execute access (POST and GET method)
- ▶ No Access

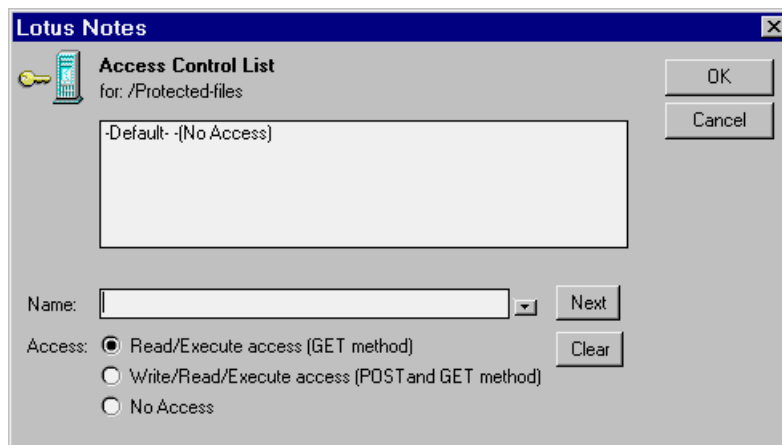


Figure 9-8 Access Control List for file protection

In the Name field, specify the user name by typing or by using the Domino Directory lookup. After assigning the appropriate access permission, click Next to apply this user to the Access Control List. To remove a user, click the name and click Clear.

9.4 Performance

The Domino server HTTP process handles Web page requests with a thread pool, consisting of a number of worker threads. To increase Domino HTTP performance, some tuning parameters are available in the Domino Directory, HTTP section.

A good way to increase the Domino Web server performance is by using the partitioning and clustering features of Domino R5, which were discussed in a previous chapter.

9.4.1 HTTP threads

Increasing the number of active HTTP threads that handle the incoming HTTP requests can increase the concurrency and thereby improve the performance of the Domino Web server.

The number of HTTP threads can be specified in the “Number of active threads” field in the HTTP section of the server document in the Domino Directory. The default setting is 40.

We recommend changing this to 96 (this value depends on how much physical memory you have in your Solaris box; 96 is a starting value for a Web server-only machine with 512 MB of memory). Creation of each thread consumes approximately 20 to 40 KB of memory, so it is not advisable to set the value beyond 100.

A method to count the number of active threads running for the HTTP process is using the UNIX command:

```
# ps -efL | grep notes | grep http | wc -l
```

The -L option of the **ps** command shows the Light Weight Processes (LWP). (This is what Solaris calls the threads). The last command “wc -l” counts the number of lines in input.

Determining the optimum number of HTTP threads requires server load tests and tuning. A good recommendation is to start by setting HTTP active threads to 10% of the estimated number of concurrent Web users. For instance, if a customer anticipates 200 users on a system, they should begin with 20 active

threads. We suggest that customers begin with a minimum of 8 threads per CPU, or two times the number of disks, and adjust upwards from there. If you increase the number of processors, the number of threads per CPU should decrease; however, generally no more than 128 threads should be specified, regardless of the number of processors.

Maximum requests over a single connection setting

HTTP is a stateless protocol. This means that HTTP does not allow for a dedicated session or connection between the browser and the server. Once a Web page has been sent to a browser, the connection is disconnected.

The HTTP 1.1 specification allows for persistent connections. The browser is able to submit additional requests without the overhead of making a new connection.

In Domino this limit is configured in the HTTP section of the server document in the Domino Directory, via the “Maximum requests over a single connection” field. The default setting in Domino R5 is 1 request; this value resulted from several benchmark tests in the Lotus labs.

Although this will cause the HTTP server to revert to HTTP 1.0 behavior and close every connection after only one request, this behavior can help avoid threads that remain idle for extended periods of time. Even though a new socket connection must be established, it will take less resources to establish the connection than if each thread remained idle.

Determining the optimum setting for “Maximum requests over a single connection” also requires server load tests and tuning.

9.4.2 Setting HTTP timeouts

There are several timeout fields that can be tuned in the HTTP section of the server document.

Input timeout

When a browser submits a connection request, a worker thread picks up the request from the connection request queue. The worker thread then attempts to read data from the socket. If for some reason the HTML request data is not present, or is incomplete, the worker thread waits for the amount of time configured in the “Input Timeout” parameter.

By default, this value is two minutes. If the thread detects new data within that time, but the data request is still incomplete, it resets the counter and waits for an additional two minutes for more data (this wait loop is unlimited). If within the wait period no new data comes in, then the socket connection is closed. If the data request is complete, the thread processes the request.

This timeout only takes effect once the thread has received and serviced a complete HTML data request, and is waiting for a new data request on the same connection. It is generally recommended that the input timeout parameter be set to the lowest value possible, which is presently one minute in length.

Output timeout

The output timeout parameter is used to determine how long a worker thread spends rendering HTML data. This applies to images, downloads, and Common Gateway Interfaces (CGIs), once the CGI has begun execution. The default value for this parameter is 20 minutes. It is used in conjunction with file downloads or responses to data requests that may take a long time to send back to the client.

Once this limit has been reached, the thread times out the socket connection. Notes Forms or Documents that run agents can also be affected by this if the triggered agent is in a loop. However, the timeout does not apply to the execution of the agent itself, but only to the socket connection with the browser. We recommend decreasing this value unless problems with file downloads are encountered. For example, in many cases this value can be lowered to between 5 and 10 minutes.

CGI timeout

This timeout is used to specify how long an HTTP worker thread waits for a CGI program to begin to execute initially. The default value for this parameter is five minutes. Once the CGI starts to execute, the CGI timeout is no longer used. At this point the output timeout parameter takes effect to control how long the thread waits for the CGI to write output to the browser.

We recommend that this timeout value be lowered, since it is reasonable to expect that in most cases the CGI will begin execution within a few seconds. For example, this timeout value can be lowered to two minutes.

9.4.3 Asynchronized Web agents

The Notes.ini variable DominoAsynchronizeAgents specifies whether agents triggered by browser clients can run at the same time.

To change the default value of 0 to 1, add the following line to the Notes.ini file:

DominoAsynchronizeAgents=1

Beginning with Domino 5.0, you can enable agents to run asynchronously through the server document. Open the server document and select the Internet Protocols - HTTP tab. Under Web Agents, enable the “Run Web Agents Concurrently” option.

Running Domino agents in parallel increases Domino Web server performance.

9.4.4 Web statistics

You can monitor the Domino Web server reading Domino statistics. Do this by issuing the Domino server console command:

```
> show stat domino
```

To redirect the command in a text file use the > operator:

```
> show stat domino >/tmp/domino.stat
```

Tip: Be careful using the blank spaces. You must put a blank before the > and no blank after, otherwise the text file will not be created.

Examining your Web server statistics will give you a lot of useful information for tuning your Web server environment.

Tip: The output of the **show stat** command conforms to DOS text file standards. If you read it on UNIX you will see a ^M at the end of the line. Use the **dos2unix** command to remove extra carriage returns and convert end of file characters in DOS format text files to conform to Solaris requirements, for example:

```
# dos2unix domino.stat domino.stat
```

9.4.5 Web stress tools

There are a lot of tools available with which to test Web performance. Good examples are Webload from Radview Software, LoadRunner from Mercury, and the ProActive Tools from G2 Associates. For more information about these products, see the following Web sites:

<http://www.radview.com>

<http://www.mercuryinteractive.com>

<http://www.g2sys.com>

9.5 Troubleshooting

Generally, HTTP process troubleshooting considers the same issues discussed in Chapter 12, “Diagnostics and Troubleshooting.” There are a few issues that are specific to the HTTP process; they are described in this section.

9.5.1 HTTP does not respond

To check if the HTTP process has hung or simply is overloaded by a lot of client requests, a good basic test you can do is telnet to the process in the right port, by default port 80.

For example, if your Domino server is running on a host named acme and listening on the default port 80, you have to run the command:

```
# telnet acme 80
```

The command output is as below:

```
# telnet acme 80
Trying 155.51.24.20...
Connected to acme.
Escape character is '^]'.
```

Now you can issue an HTTP command, for example **get**:

```
Trying 155.51.24.20...
Connected to iena.
Escape character is '^]'.
```

```
get
<HTML>
<!-- Lotus-Domino (Release 5.0.2b - 16 December 1999 on Solaris Sparc) -->
<HEAD>
</HEAD>
<BODY TEXT="000000" BGCOLOR="000000">

<FORM><DIV ALIGN=center>
```

In this case the **get** command receives an answer from the HTTP process; if HTTP was hanging the **get** command would not receive any responses.

Tip: This technique can be implemented also for the other Domino Internet processes, like IMAP, LDAP, and POP3, by choosing the appropriate port number (for example, 143 for IMAP) and the appropriate protocol command (for example, **hello** for IMAP).

A useful tool to trace the activity of HTTP during the hang is the truss tool, described in detail in Chapter 12.

9.5.2 Using the tell command

Domino R5 introduces a new console command that helps in troubleshooting if HTTP hangs on R5. This command is **tell http Show Thread State**.

When entered at the Domino console, this command displays the current status of each active thread, and which URL, if any, the thread is processing.

Following is a sample output for three threads. The first two threads are idle; the third thread (0xf9) is processing the URL

```
GET /reference.nsf/ Refresh?OpenAgent HTTP/1.0
```

```
> tell http show thread state
04:37:09 PM HTTP Thread State: Thread: [fb] State: [Worker waiting for work]
Other Info:
04:37:09 PM HTTP Thread State: Thread: [fc] State: [Worker waiting for work]
Other Info:
04:37:09 PM HTTP Thread State: Thread: [f9] State: [Worker processing request]
Other Info: GET /reference.nsf/Refresh?OpenAgent HTTP/1.0
```

If the HTTP process is in a hung or partially hung state, this command can be used to determine if a particular thread has been processing the same URL for too long. If the thread is still processing the same request or URL for more than a few minutes, then the thread is likely hung. You can check this by repeating the command after a few minutes.

In many cases, if the HTTP task is hung, the Domino administrator can attempt to shut the HTTP server task down, but the task does not always shut down gracefully. In Domino R5, when an administrator issues the command **tell http quit**, if HTTP is waiting for a hung thread to complete during shutdown, HTTP outputs this thread ID and the URL it is working on to the console. For example:

```
> tell http quit
04/28/99 04:37:51 PM HTTP Waiting For Thread: Thread: [f9] State: [Worker
processing request] Other Info: GET /reference.nsf/Refresh?OpenAgent HTTP/1.0
```

This information can be used to determine the hung thread, and which URL the thread is processing. Match this information with the NSD output file, which is described in detail in Chapter 12.

This is similar to the use of the req*.log files. The thread ID can be correlated against the req*.log file that pertains to that thread.

9.5.3 Bindsock issue

If you take off the suid bit in the bindsock process you will have a socket bind error message in the log and the Domino Web server will not start. You will see these error messages in the Domino console and in the log.nsf database:

```
12/20/99 04:12:20 PM HTTP Socket bind error, hostname/ip perignon.iris.com
12/20/99 04:12:20 PM HTTP server: Could not bind port 80. Port may be in use
12/20/99 04:12:20 PM HTTP Web server shutdown
```

We have experienced this kind of problem from time to time, when the UNIX administrator worked with the Domino administrator and made a `chmod 777 *` command in the Domino program directory, removing the suid on the bindsock file.

Note: You must be very careful changing the default file permissions in the Domino program directory.

9.5.4 HTTP thread debugging

Additional diagnostics for the Domino HTTP process are available, and can be enabled when troubleshooting HTTP problems.

A request log file can be created for each worker thread by placing the parameter "debugthreadlogging on" in the httpd.cnf configuration file. When this is enabled, a file is created for each active thread, with information about each request processed appended to the file as requests are made to the server (roughly 10-15 lines per request). These files can be extremely useful to pinpoint causes of HTTP crashes or hangs.

As an alternative to placing "debugthreadlogging on" in the httpd.cnf, administrators can enter the following command at the server console:

```
>tell http debug thread on
```

This dynamically sets the thread logging debug flag, and the server begins to create thread logs immediately. However, this debug flag remains in effect only until the HTTP server is restarted. This method of turning on debug does not place the parameter in the httpd.cnf file.

The created files are named req###.log, where ### is the thread ID for the active thread, and they are written to the Domino data directory. Example 9-1 on page 248 is a sample output for one request from a req file, where "###" is the thread ID as taken from the nsd, lwp on Solaris. For instance, req110.log corresponds to the lwp-id 110 from the nsd.

Example 9-1 Sample output for one request from a req file

```
*** Start Request 1750
*** Parsing Request 1760
*** Read socket 27, Status 260, 1760
GET /log.nsf HTTP/1.0
Connection: Keep-Alive
User-Agent: Mozilla/4.72 [en] (WinNT; I)
Host: iena
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
Accept-Encoding: gzip
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
*** Finished Parsing Request 1760
*** Calling Redirection 1760
*** Calling Internotes 1760
*** Wrote socket 27, Status 494, 1790
HTTP/1.1 401 Unauthorized
Server: Lotus-Domino/5.0.8
Date: Mon, 06 Aug 2001 22:56:18 GMT
Connection: close
Content-Type: text/html; charset=US-ASCII
Content-Length: 224
Expires: Tue, 01 Jan 1980 06:00:00 GMT
WWW-Authenticate: Basic realm="/"
Pragma: no-cache
<HTML>
<HEAD>
<TITLE>Error</TITLE></HEAD>
<BODY TEXT="000000">
<H1>Error 401</H1>You are not authorized to perform this operation
<P><HR><ADDRESS><A HREF="/">Lotus-Domino Release 5.0.8
(France)</A></ADDRESS></BODY>
</HTML>
*** Internotes Processed the request 1790
*** Before Cleanup Memory 1790
*** Request Processed 1790
*** Closing Socket 27 1800
```

These req*.log files do not contain a date/time stamp, so they must be used in conjunction with Domino logging (DOMLOG.NSF or Access logs). However, each line of the logged request displays the number of milliseconds since the HTTP process last started (see bold number in "Start Request" line). This allows you to determine the amount of time that each phase of the request process takes.

Note: Use these variables for debugging only. They have a significant impact on server performance when they are enabled.

It is also useful to have an NSD output to check what the thread was doing at crash/hanging time. See Chapter 12, “Diagnostics and troubleshooting” on page 305 for information on how to use the NSD tool.

Tip: To get only the HTTP thread information, use NSD with the HTTP PID as parameters.

9.5.5 Memory leaks

The UNIX command **ps -efly** can be used to show the memory consumed by a UNIX process.

To check the memory used by the HTTP process you can use:

```
# ps -efly | grep http
```

The memory values are indicated in the RSS (Resident Set Size) and SZ (Set Size) columns. The RSS value is the resident size of the memory; SZ value is the total size of the process in virtual memory.

Normally the RSS value is less than the SZ value.

These values should be stable. If they continue to grow that could be a symptom of a memory leak.

A memory leak in the HTTP process can be caused by various things. In the majority of cases it is caused by the Web applications running on the Domino server, like Web agents written in Java.

Tip: To make a memory dump of all the Domino processes, as the Notes user go to the Domino data directory and at the UNIX prompt issue the command **server -m**. This will create an ASCII file called **memory.dmp**. Send this file to the Lotus Support team for help with the troubleshooting.

9.6 Domino R5 console tell commands

Lotus Domino R5 has additional **te11** commands that can be used for the HTTP process. The new commands are:

- ▶ `tell http show users`
- ▶ `tell http show thread state`
- ▶ `tell http restart`
- ▶ `tell http show security`
- ▶ `tell http show virtual servers`

tell http show users

This command can only be used if the server is configured to use session-based tracking for the Web. Session tracking is a feature of session-based authentication. To enable it, edit the server document in the Domino Directory. In the Internet Protocols section, select Domino Web Engine. By default, the entry for “Session authentication” is disabled. Select “Enabled” to allow the HTTP task to report on authenticated users. This command will show the User Name, IP address and the time of expiration (which is 30 minutes by default). This will only reflect users who are authenticated, and cannot be used to track anonymous users.

tell http show thread state

This command will list the current state of each active thread (as well as the accept thread and logger thread). If the thread is processing a request, the output of this command will indicate the URL being processed.

tell http restart

This will cause the HTTP task to shut down and reload. This is the equivalent if **`tell http quit`** followed by **`load http`**. This command is valid for the other Domino processes, too.

tell http show security

This outputs current status on the use of SSL for the server and each virtual server.

tell http show virtual servers

This outputs the current configuration for virtual servers.

9.7 Virtual servers and host

If you are an Internet Service Provider (ISP) or corporate intranet administrator who provides services to multiple customers, you can set up virtual servers on a single Domino Web server. A single Domino Web server can then host several Web sites. Using virtual servers allows you to maintain separate sites without incurring the expense of additional hardware and software.

You can configure each site in Domino with its own IP address, default home page, customized Web server messages, and HTML, CGI, and icons directories. The Domino data directory, however, is not individually configured for each virtual server; it is shared by all virtual servers.

The difference between a virtual server and a virtual host is that virtual servers have different IP addresses and different hostnames, while virtual hosts use the same IP address but different hostnames.

9.7.1 Network setup

You have to change the network setup of your Solaris box before you can configure Domino virtual servers or hosts.

Virtual server

For Domino virtual servers you have to add additional IP addresses to be used for each virtual server.

Here is an example using a single network card and the IP alias mechanism on Solaris.

First create “/etc/hostname.device:n” files that contain the name of each virtual server, where device is the device name of the NIC (Network Interface Card) and n is a number that increments for each filename.

Note: In Solaris 7 and 8 the device name usually is hme0; for the first network card, hme1, and so on.

For example, if you want to create three Domino virtual servers, these steps are necessary:

1. The file /etc/hostname.hme0 file should already exist for the hostname of the physical server and contains the name server1.
2. Create the file /etc/hostname.hme0:1 containing the server name server2.
3. Create the file /etc/hostname.hme0:2 containing the server name server3.
4. Then issue the following commands (only if hme0 unplumbed):

```
# /sbin/ifconfig hme0 plumb  
# /sbin/ifconfig hme0:n <IP_address>up
```

where n corresponds to the number of the /etc/hostname.hme0:n files.

For example: **/sbin/ifconfig hme0:1 9.3.187.210 up**

Note: If you want to disable the IP address, type the command:
`/sbin/ifconfig hme0:n down`

If you don't use Domain Name System (DNS), you have to put the new IP addresses in the `/etc/hosts` file:

```
9.3.187.210 server2
9.3.187.211 server3
```

If you use DNS, ask your DNS administrator to add new IP hostname aliases into the DNS.

Note: Use DNS instead of hostfiles whenever possible!

Virtual host

For the virtual hosts it is not necessary to configure new IP addresses. You only have to add the new hostnames in the `/etc/hosts` file or DNS system.

For the `/etc/hosts` file:

```
9.3.187.210 server1 server2 server3
```

Ask your DNS administrator to add the new hostnames in the DNS.

9.7.2 Create virtual server or host

If you want to create a virtual server or host, in the Domino Directory select the Domino server and choose Actions -> Web -> Create virtual servers from the menu bar.

Now you will be asked whether you want to create a virtual host or a virtual server, as shown in Figure 9-9.

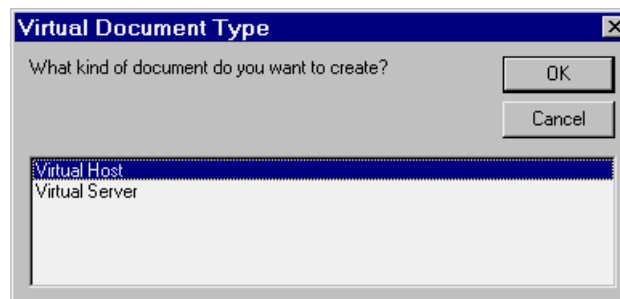


Figure 9-9 Creating virtual host

Choose Virtual Host. Creating a virtual server is pretty much the same, except you will be asked for the IP address instead of the hostname.

On the Basics tab, enter the hostname of your added virtual host.

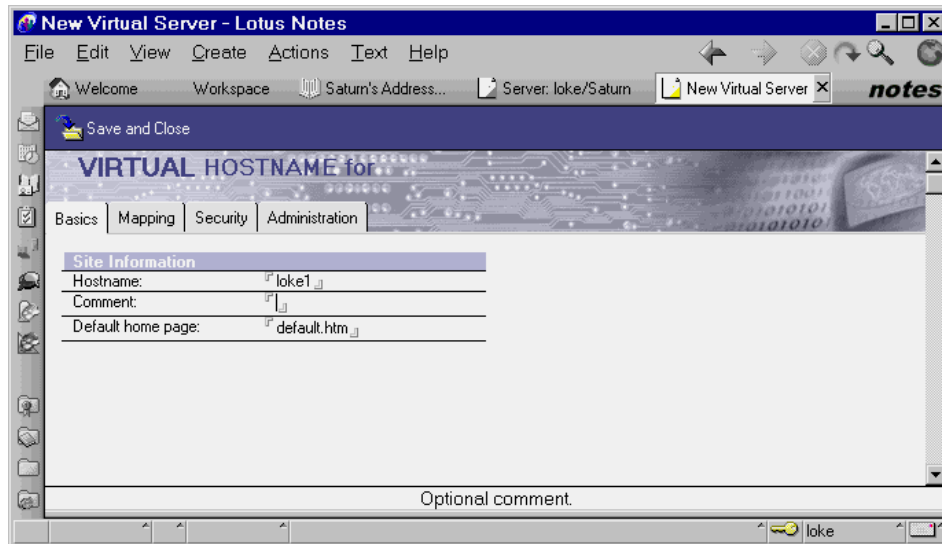


Figure 9-10 Virtual host

On the Mapping tab, specify the path names mapping to the HTML directory, the Icon directory, the CGI directory, and the home URL, like a Domino Web server configuration. This tab is the same for both server types.

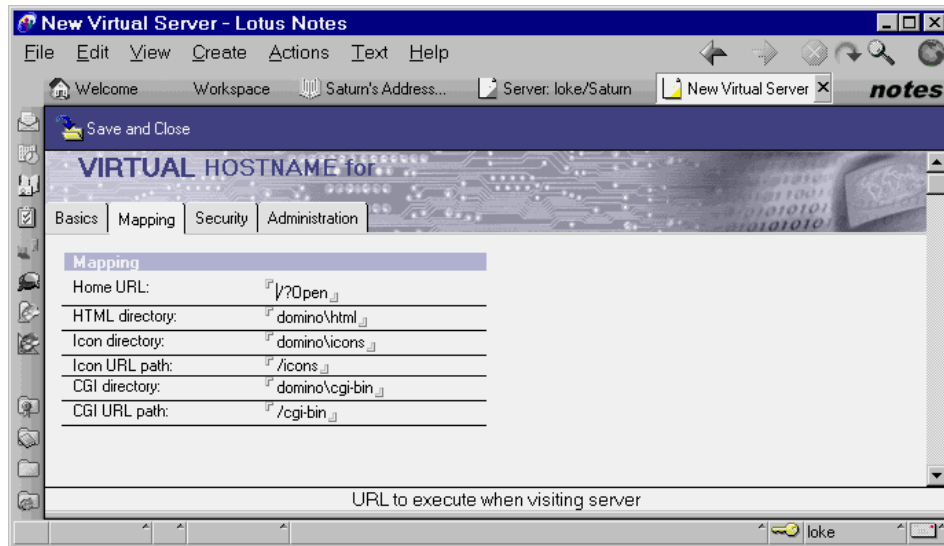


Figure 9-11 Virtual host mapping

The Security tab lets you make some security settings for your virtual servers. You can decide if Name and password and/or anonymous authentication can be used.

You can also customize the SSL settings. For more information on SSL, see “Invoking SSL on Your Domino server” in Chapter 4 in *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341.

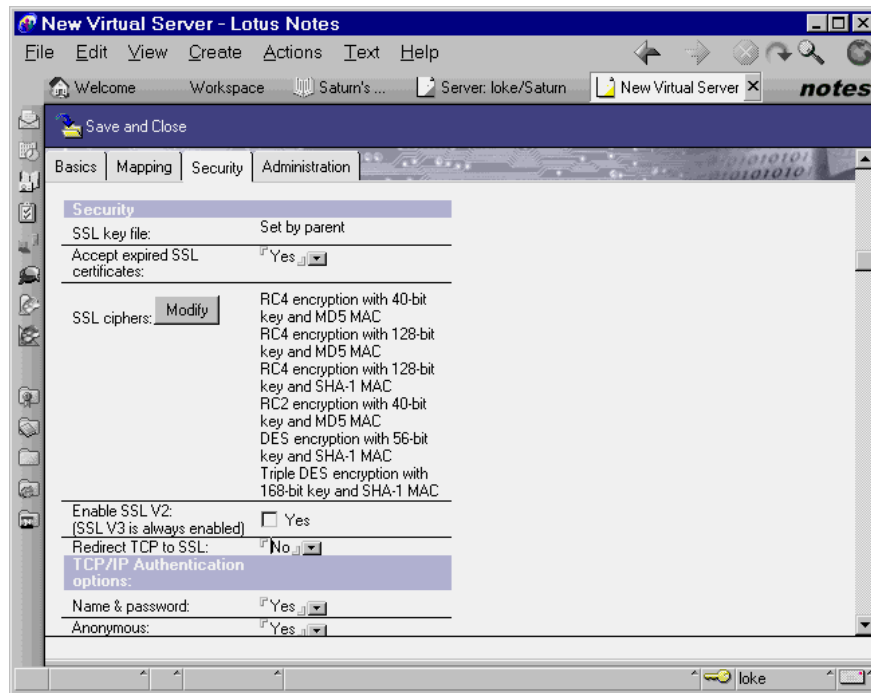


Figure 9-12 Virtual host security

9.7.3 Create URL mapping and redirection

There are three different types of URL mappings. Depending on your choice, you will get three or four tabs to configure the mapping.

1. URL-to-URL mapping enables you to define an alias name for URL paths. For example, you could map /MyPictures to /images. Figure 9-13 shows URL-to-URL Mapping.

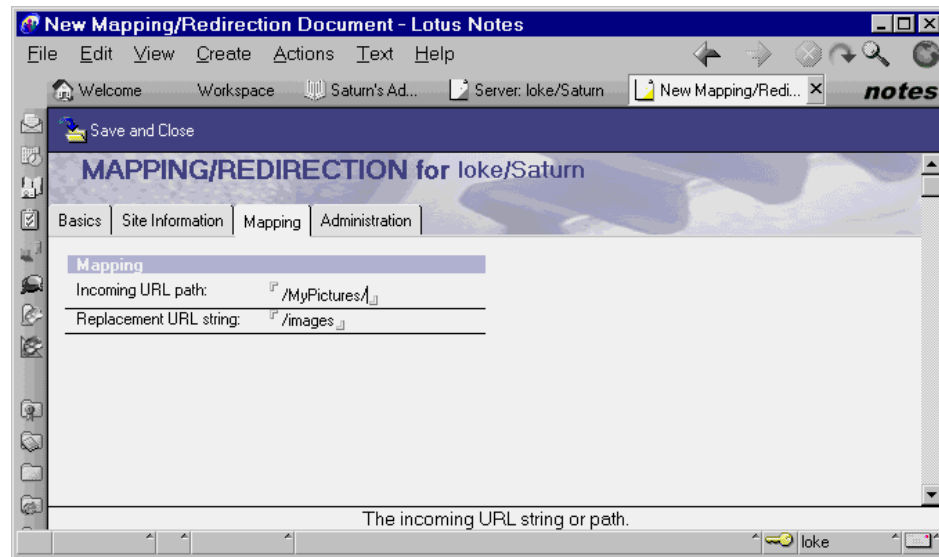


Figure 9-13 URL mapping/redirection document

2. URL-to-Directory mapping enables you to specify which URL path should be mapped to which real directory on your server. For example, if you have all the images you are using in your Web pages in a directory /web/images, you have to create a directory mapping /web/images to /MyPictures to be able to access these pictures through the Internet. Figure 9-14 on page 257 shows URL-to-Directory mapping.

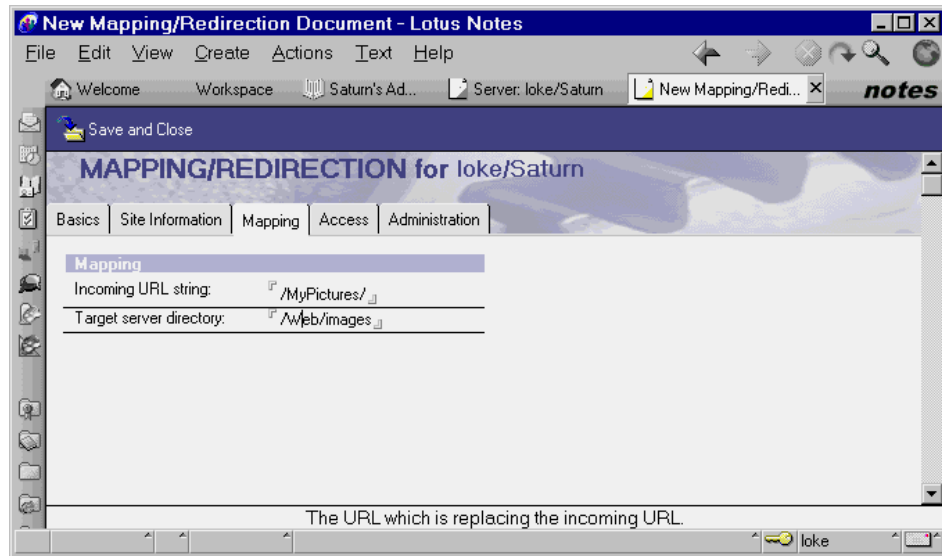


Figure 9-14 Changing URL mapping/redirection to a directory

3. Redirection URL-to-URL. Using this, you can move pages to a different server without making the old URL invalid. Figure 9-15 shows Redirection URL-to-URL.

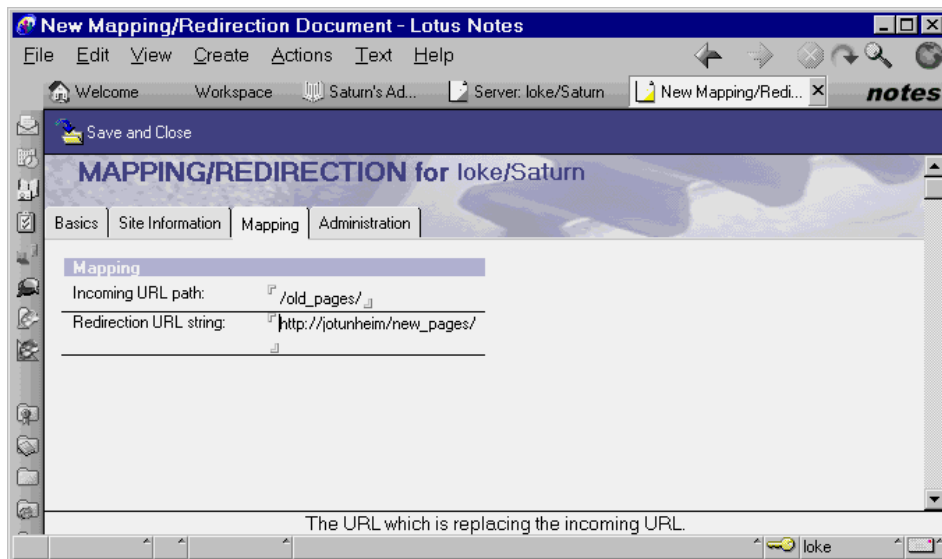


Figure 9-15 Redirecting URLs to another URL

For each choice, specify in the Site Information tab which virtual server is affected by this mapping. On the Mapping tab, specify the actual mapping.

If you have defined a URL-to-directory mapping, you will also have to specify if your data can only be read or if it should be executable.

9.8 The Internet Cluster Manager (ICM)

Domino clustering in R5 extends failover and load balancing capabilities to Web browser clients (HTTP and HTTPS). It is the Internet Cluster Manager (ICM), a new Domino R5 server task, that's behind these new capabilities.

The ICM serves as a liaison between the HTTP clients and the HTTP servers of a Domino cluster. The HTTP clients direct requests for a database to the ICM. The ICM maintains information on the availability of the Domino servers in the Domino cluster, and also maintains information about the distribution of databases on the servers. The ICM determines the best server to receive a particular client request and directs the request to that server.

The ICM can do the following:

- ▶ Monitor back-end Domino servers for availability
- ▶ Check the Domino HTTP Web service for availability
- ▶ Disallow any new connections to servers out of service
- ▶ Provide failover (direct clients) to the best available server
- ▶ Provide load balancing by setting availability thresholds for your Domino servers
- ▶ Support “virtual IP addresses” and map ports
- ▶ Provide content routing for your clients

9.8.1 Configuration

An ICM process is dedicated to a single Domino cluster. Therefore, if you have two clusters, you must have separate ICMs for each cluster. The ICM needs to be in the same domain as the Domino cluster because the ICM always uses the local copy of the Domino Directory.

You can configure the ICM in several ways. First, you can have the Domino server running the ICM dedicated to that purpose and configured to run outside of the cluster.

The server should not contain any databases other than those necessary for server operation, and should only run the basic set of server tasks. Configuring the ICM in this way makes it more reliable because there are fewer activities performed on the server that could interfere with performance and lead to server failure.

You can improve the availability of the ICM by configuring more than one ICM to handle user requests. Then, if one ICM becomes unavailable, the other remains available so that client requests are still handled. You can run multiple ICMs on a single physical machine by using Domino partitioned servers.

In addition, you can run each ICM on its own Domino server on separate physical machines. Typically, you configure the ICMs with the same hostname in the Domain Name Server (DNS). This way, if one of the ICMs fails, the other ICM takes over without affecting users.

Note: You can use an OS cluster to provide failover support for the ICM running outside of the cluster. Use the Sun Cluster 2.2.

Of course, you can also configure the ICM to run on one or more of the Domino servers in the cluster. Be sure that the servers can handle the added traffic that the ICM generates. Again, the availability of the ICM improves by running more than one ICM for the cluster.

Configuration steps

You configure the ICM by making entries in the Internet Cluster Manager section of the server document. You can also set up a separate IP address for the ICM. You can then start the ICM.

To configure the ICM:

1. In the Domino Administrator, click the Configuration tab.
2. Expand Server and click All Server Documents.
3. Select the server document for the server on which you want to run the ICM, then click Edit Server.
4. Click the Server Tasks - Internet Cluster Manager tab.
5. Complete the following fields:
 - Cluster name
Name of the Domino cluster the ICM will service. (This is only necessary if the ICM is run on a server outside the cluster.)
 - Get Configuration from
This field lets you specify a different server document to get ICM

configuration information from. This is helpful if you want to set up multiple ICMs to share the same configuration. If you select to get the configuration from “another server document,” you can enter the name of the server in the “Obtain ICM configuration from” field that appears.

- ICM hostname
The fully qualified name of the host that clients should use to communicate with the ICM. This can be the registered DNS name or the IP address. (The Domino HTTP server uses this field to create URLs that reference the ICM. If this field is blank, the HTTP server will not be able to generate URLs that refer to the ICM.)
- TCP/IP port number
The port number for the ICM to use. (If you are running the ICM on the same server as the HTTP server, you must avoid address and port conflicts. If you do not give the ICM its own IP address, be sure the port number the ICM is using is different from any of the other port numbers you use on the server.)
- TCP/IP port status
To enable HTTP communication with the ICM, choose Enabled. To disable HTTP communication with the ICM, choose Disabled.
- SSL port number
Enter the port number to use for SSL. (If you are running the ICM on the same server as the HTTP server, and you do not give the ICM its own IP address, be sure the SSL port number is different from any of the other port numbers you use on the server.)
- SSL port status
To enable HTTPS communication with the ICM, choose Enabled. To disable HTTPS communication with the ICM, choose Disabled.

Figure 9-16 on page 261 shows an example of ICM configuration.

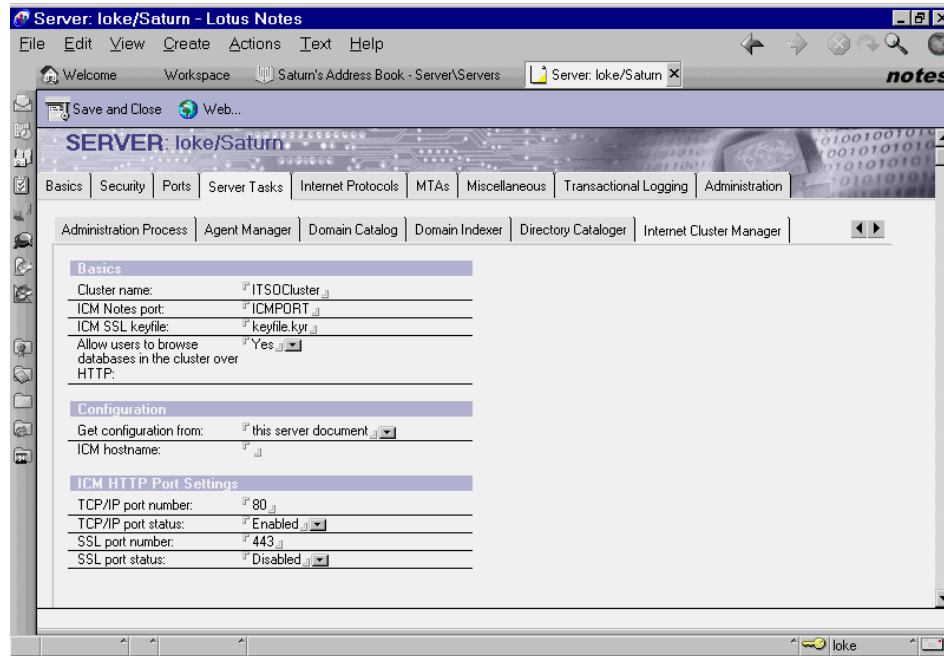


Figure 9-16 Internet Cluster Manager (ICM) configuration document

You should enter the ICM configuration information for all servers in the cluster, even for servers not running the ICM server task. This is necessary because the Domino HTTP server uses the ICM configuration from its own server document to determine how to direct HTTP clients to the ICM when appropriate.

If the ICM runs on the same system as the Domino HTTP server, you can configure either the ICM or the HTTP task to run on separate ports or to run with different IP addresses in order to avoid conflicts. The recommended approach is to assign the ICM its own IP address (using the ICM hostname field described previously). You can also share IP addresses and assign different ports to the ICM and Domino HTTP server.

To start the ICM, append the ICM keyword to the ServerTasks setting in the Notes.ini file. For example:

```
ServerTasks=ROUTER,REPLICA,ADMINP,CLDBDIR,CLREPL,HTTP,ICM
```

You can also start the ICM process from the Domino server console, as a normal Domino process, with the command:

```
> load icm
```

To stop it, use:

```
> tell icm quit
```

It is not recommended to stop a Domino process using the **kill** UNIX command at a UNIX prompt; this can produce a “Freezing all server threads...” issue. See Chapter 12 for more details. Use the **kill** command only if the Domino process does not respond to a **tell quit**; restarting the Domino server in this case should be necessary.

9.8.2 Statistics

To collect the ICM statistics you can use the Domino console command:

```
> show stat ICM
```

You can use the ICM statistics to tune your cluster environment.

Following is the list of the ICM statistics and their meanings:

- ▶ **ICM.AvailabilityIndex.ServerName**
A measure of a server's availability. Zero (0) indicates no available resources, 100 indicates complete server availability.
- ▶ **ICM.Command.Total**
The number of URL commands the ICM received.
- ▶ **ICM.Command.Unknown**
The number of URL commands the ICM did not recognize.
- ▶ **ICM.Receive.Error**
The number of times the ICM could not process a client request because of a communication problem between the client and the ICM.
- ▶ **ICM.Command.Redirects.Successful**
The number of times the ICM successfully redirected a client URL request to a cluster member.
- ▶ **ICM.Command.Redirects.Unsuccessful**
The number of times the ICM could not redirect a client URL request to a cluster member.
- ▶ **ICM.Command.Redirects.ClusterBusy**
The number of times the ICM received a client request when all servers were BUSY.
- ▶ **ICM.Requests.Per1Hour.Total**
The number of HTTP requests the ICM received in the past hour.
- ▶ **ICM.Requests.Per1Minute.Total**
The number of HTTP requests the ICM received in the past minute.

- `ICM.Requests.Per5Minutes.Total`
The number of HTTP requests the ICM received in the past 5 minutes.

9.8.3 Troubleshooting

When troubleshooting an ICM crash, you should enable the following debug variable in the `Notes.ini` file:

```
ICMDebugLevel=n
```

where `n=[1-3]` for the 3 different levels of logging for the ICM process (1,2,3), and level 3 is the most verbose. Information such as routing expense, URL requests, and server redirection are logged with this debug. This will allow you to determine if a specific URL is causing a crash, or why a particular member of the cluster is not being routed to via ICM.

You must use this variable with caution; a lot of debug information can be displayed at the Domino server console. Use this only for troubleshooting an ICM crash, and start with the less verbose level 1.

The same considerations discussed in Chapter 12 are valid for the ICM process.

9.9 Domino and Java

At the time of this writing Lotus Domino R5 includes a Java Virtual Machine (JVM) based on Sun Microsystem's JDK. The JVM is automatically installed in the Domino program directory.

The HTTP server task always loads the JVM when the HTTP task is started.

9.9.1 Using a different JVM

It is possible to use another JVM in substitution for the JVM shipped with Domino. To do this you have to integrate the new JVM in the Domino installation using a UNIX shell script, called `notesjre`. You can find this script in the Domino binary directories.

At the time of this writing the `notesjre` script can be used only to integrate JVM 1.1.x. Lotus is currently working on a version that will be valid for JVM 1.2.x and above, where the JDK/JRE directory structure changed radically from previous releases.

The `notesjre` script can be started with different options:

```
notesjre [-n:j:srt:h]
```

- n notesdir Set Notes directory to notesdir
- j jredir Set JRE directory to jredir
- s Symbolically link JRE into NOTESDIR (default is to copy files)
- r Restore JRE from backup dir (mutually exclusive to -m)
- t threads Thread type to use: native, green (default is native)
- v Display version information
- h Display usage

You should start the script as the Domino user; if you have a problem you can use the root account, too. If you start the script without options, notesjre shows a brief description of how it works:

► **# notesjre**

This is the default usage for an end user and causes the JRE from the default system path to be copied into the default location of a Notes installation. (The existing JRE which shipped with Notes will be placed in a backup, only if not already moved there, subdirectory which can be restored at any time by using the -r option.)

► **# notesjre -m -n /opt/lotus/notes/latest/ibmpow -j /usr/lpp/JRE**

Sets Notes directory (NOTESDIR) to /opt/lotus/notes/latest/ibmpow. Sets JRE directory (JREDIR) to /usr/lpp/JRE. Use default thread model (native_threads). Checks for the existence of the backup directory NOTESDIR/jrebackup, creates if needed. Moves any existing JRE files from NOTESDIR to NOTESDIR/jrebackup. Creates links for JRE files from NOTESDIR to proper spots in JREDIR.

A useful option is the restore option -r, which restores the original Domino JVM:

notesjre -r

Tip: Do not use the notesjre script twice—otherwise the jreback directories, where the original JVM is stored, will be overwritten by the new JVM files. If you want to restart the notesjre script you have to first restore the original JVM.

To check which Java version is installed in your Solaris System, issue the command:

```
# java -fullversion
java full version "Solaris_JDK_1.1.6_03"
```

9.9.2 Java servlets

A servlet is a Java program that runs on a Web server in response to a browser request. Servlets for Domino must conform to the Java Servlet API Specification, an open standard published by Sun Microsystems, Inc.

Configuring

On a Domino R5 server, Java servlet support is disabled by default. In order to enable Java servlets, edit the server document and go to the Domino Web Engine tab, and find the section labeled “Java Servlets.” Set the appropriate value for the field “Java servlet support.” In R5 there are 3 options:

- ▶ None.
- ▶ Domino Servlet Manager (which initializes the Domino JVM and starts the servlet manager).
- ▶ Third party Servlet manager (which initializes the Domino JVM only). In order to use a third party servlet manager, one must install the appropriate software (such as IBM WebSphere) which will in turn place lines in the HTTPD.CNF file to allow the servlet manager to plug in to the Domino HTTP server.

When the servlet manager is enabled (either the Domino Servlet Manager or a third party servlet manager), several lines similar to the ones following should appear in the domino.cnf file (which is located in the data directory):

```
# Java Servlet Settings
ServerInit servlet:ServletInit /export/home/s5020/notesr5
Service /servlet/* servlet:ServletService*
ServerTerm servlet:ServletTerm
```

Note: The domino.cnf file contains the HTTP data configurations. Do not edit it directly; use the Domino Directory to change the HTTP configuration.

Running

The basic steps to run a servlet in R5 are as follows:

1. In the “Servlet URL Path” field, enter the URL path you wish to use to indicate that the resource is a servlet (the string “/servlet” is the default).
2. Create a directory under the data/domino directory (for instance domino/servlets) where you wish to store your servlets.
3. Edit the “Class Path” field to include the location of your specific servlet (this replaces the JavaUserClasses Notes.ini entry in R4.6). You can specify .jar and .zip files in this field.
4. Copy the class files to the data/domino/servlets directory.

5. Issue the server console command **tell http restart** to reload the HTTP server. In your Web browser, enter a URL that contains the servlet name (without the file extension), such as:

`http://hostname/servlet/HelloWorldServlet`

Note: The addition of any servlets to the servlet directory will require a restart of HTTP before the servlet manager will recognize the new servlet.

At the time of this writing Domino R5 supports Sun Microsystems' Java Servlet Development Kit (JSDK) release 2.0.

9.10 Domino log and analysis tools

Domino R5 makes logging even easier for Internet service providers (ISPs), as well as the rest of us. Domino R5 can now create text files that include the IP address or host name of the server that the user requests. This way, you can more easily use the logs to create statistics for virtual servers. To use this feature, you must enable the "Extended log format" for the access log file in the server document.

To create separate statistics for virtual servers, analysis tools still need to sort the entries in the log file according to the different virtual servers' IP addresses or host names.

9.10.1 Domino Web log

To set up logging on your Domino server, you simply enable one of the logging methods in the HTTP section of the server document in the Public Address Book. (Because logging is very server-intensive, it is disabled by default.)

If you enable logging to domlog.nsf, the database is automatically created the next time you start the server. If you enable logging to text files and specify a directory for the files, Domino automatically creates the access log and error log files.

Notice that you can select the format for the access log files (Common or Extended Common) and the time format (LocalTime or GMT). Remember that the Common format records only access information, and the Extended format tracks access, agent, and referrer information in the access log file. You can then specify different names for the log files.

Figure 9-17 on page 267 shows the logging fields in the server document.

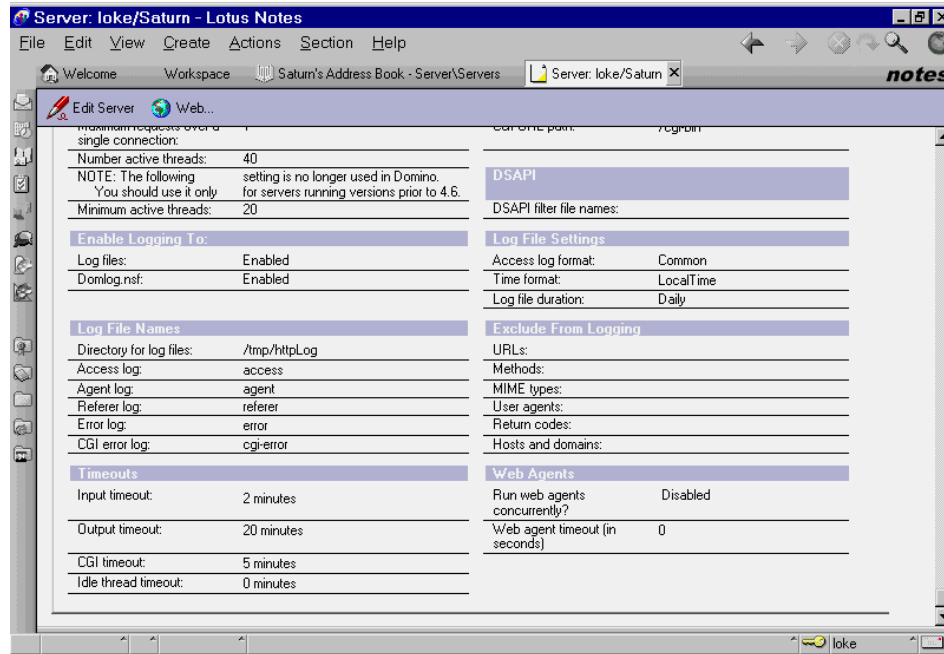


Figure 9-17 Domino Web logs

Logging Fields

With Domino R5, you can now specify whether you want Domino to create new log files daily, weekly, monthly, or never. In previous releases, Domino created a new log file at midnight each night. The log file duration applies to all log files on the server. In addition, only one log file is maintained per Web server, including servers set up as virtual servers. The name Domino gives to the log file depends on the duration settings and the file names you specify in the server document.

In the “Exclude from Logging” section, you can prevent logging for specific types of requests. For example, let’s consider that you don’t want to log image requests on your server. So, you enter *.gif in the URLs field and image/gif, image/jpeg, and image/bmp in the MIME types field.

You can also prevent logging for:

- ▶ Specific HTTP methods
- ▶ User agents
- ▶ Status return codes
- ▶ Hosts and domains

9.10.2 Text file analysis

When you log to text files, the data is in a very simple format that you can easily import into a third-party application. Several vendors are starting to specialize in Domino log analysis.

In general, you import your Domino log files into their package, and then they output a wide variety of summary reports. Most of the vendors use standard database technology, such as a flavor of SQL, so you can also create your own reporting.

The following Business Partners currently provide tools for log analysis based on Domino log files. They are listed here in alphabetical order, along with their product names and Web site addresses. Contact the vendors directly for more information.

Table 9-1 Third party log analysis tools

Business Partner	Log analysis product	Web address
MarketWave	Hit List Pro	www.accrue.com
Sane Solutions	NetTracker	www.sane.com
WebTrends	WebTrends Suite for Lotus Domino	www.webtrends.com

If you are looking for free software, you can use the Analog reporting tool to analyze the log files produced by the Web server. You can find this free tool at:

<http://www.statslab.cam.ac.uk/~sret1/analog>

We were able to use it successfully in a test with the log files produced by Lotus Domino.

9.10.3 Domino Log database analysis

When you enable logging to the Domino Log database, Domino automatically creates the database using the template domlog.ntf. The basic design of the database includes one form for log entries and one view for displaying them, called Requests. The Requests view shows all records in the order that they were created. To analyze the entries in your Domino Log database, you can either use a Notes tool or one of several solutions from Lotus Business Partners. You can customize the database with additional views, create agents to notify you when specific events occur (such as, when a certain number of unsuccessful login attempts occur), or modify the database to generate reports.

Or, you may instead want to use a Domino-based Business Partner solution that's already done this work for you. Typically, these solutions query the Domino Log database, convert the hits into a more meaningful format, and then produce separate, structured log databases from the data. The following Business Partners currently offer log analysis tools based on the Domino Log database.

Table 9-2 Third party Domino log analysis tools.

Business Partner	Log analysis product	Web address
EntreVision	EntreWeb Perspective	www.entrevision.com
Unipower Systems Limited	Domi-Know.Stats	www.domi-know.com
Workflo Systems	Domino Log Analyzer	www.wfs.com

9.11 Summary

In this chapter we discussed using your Domino server as a Web server. The topics we covered were configuration, performance, security, and troubleshooting, with special emphasis on Solaris platform-specific issues.



Enterprise integration

This chapter discusses Domino Enterprise Connection Services (DECS), Lotus Enterprise Integrator (LEI) and ODBC drivers for Solaris. DECS and LEI are two of the many tools available that allow you to integrate external data with Domino. ODBC is an alternative method for accessing smaller amounts of data from Domino.

We will focus on the installation and configuration of DECS and LEI on the Solaris platform. For more detailed information on using DECS and LEI see the IBM Redbook *Lotus Domino R5 Enterprise Integration: Architecture and Products*, SG24-5593.

10.1 Domino Enterprise Connection Services

Domino Enterprise Connection Services (DECS) is an add-in task which is forms-based for easy setup. DECS provides the capability to integrate live data from enterprise systems (DB2, Oracle, Sybase, EDA/SQL, and ODBC).

The DECS add-in server task waits for user-initiated events, then passes them to the Domino Extensions Manager, which makes the query on behalf of the user. Results are then transparently returned to the user as if the Domino server had processed the transaction.

The following major features were introduced in Domino R5:

- ▶ Multi-value support for one-to-many data relationships. The expand/collapse metaconnector from Lotus Enterprise Integrator was added to DECS.
- ▶ Automatic reconnect if a connection is dropped by the enterprise system.
- ▶ Better stored procedure support.
- ▶ Support for ERP and transaction connectors. This version of DECS can be used with the SAP R/3, PeopleSoft, BEA Tuxedo, JD Edwards and Oracle Applications Lotus Connectors.
- ▶ Lotus Connectors LotusScript Extensions (LC-LSX) and RDBMS connectors have been updated to support connection pooling when using LotusScript.
- ▶ Scheduler for automatic startup and shutdown of DECS activities.
- ▶ Ability to run more than 128 concurrent activities.
- ▶ An update to the ODBC Connector for compatibility with ODBC 3.5 and to allow access to Microsoft SQL Server 7 data sources.
- ▶ User-controlled subfield key ordering.
- ▶ Improved functionality for domain searches and doclinks.
- ▶ The DECS Initialize Keys functionality has been modified to allow key fields of the NUMERIC data type.
- ▶ Stored procedure browsing and selection from the Connection document.

10.1.1 Installation

DECS is shown as an option during the initial installation of a server. By checking this box you are telling the server to configure the Notes.ini for DECS. If you have a server which is already up and running on which you would like DECS installed, you only need to change a few things in the Notes.ini. To configure the server to load DECS on startup, add the `decs` task to the `servertasks=` line in the Notes.ini:

```
Servertasks=Replica,Router,Update,Stats,AMgr,Adminp,Sched,CalConn,Event,http,decs
```

Also add the following line at the end of the Notes.ini:

```
EXTMGR_ADDINS=decsext
```

Tip: If the server returns an error such as “DECS: DECS Server Extension Manager library is not being initialized. Make sure the DECS Server is properly installed and the line 'EXTMGR_ADDINS=libdecsext.so' is in the Notes.ini file” and the EXTMGR_ADDINS= line appears to be correct, the path might not reflect the binary directory. In this case, either add the binary directory to the path of the Notes user (recommended), *or* specify the full path to the decs extensions. Typically, this would be the following line:

```
EXTMGR_ADDINS= /opt/lotus/notes/latest/sunspa/libdecsext.so.
```

10.1.2 Running DECS

To load DECS on the Domino server, simply type the following at the server console:

```
> load decs
```

The server will respond with the message, “Connection Server Started” along with the current date and time.

To shut down the DECS server, type the following at the server console:

```
> tell decs quit
```

The server console responds with the message, “Connection Server Shutdown Complete” together with the current date and time.

Note: DECS will work on a Domino partitioned server, but only *one* instance of DECS is allowed. Multiple instances on other partitions will fail.

10.2 Lotus Enterprise Integrator

Lotus Enterprise Integrator (LEI) is a “middleware” product that connects and transfers information between Domino Connector enterprise sources on a scheduled or event-driven basis.

At the time of this writing the latest version of LEI is 3.1a. It consists of three components:

1. A Domino application, called the Lotus Enterprise Integrator Administrator Database. System managers create Domino form documents, called Activities, that serve as the instruction set for the second product component.
2. The Lotus Enterprise Integrator server. The LEI server processes instructions, connecting to external data sources and moving data according to Activity-defined conditions.
3. The third component is an LEI server Log, which records data transfers and error messages.

10.2.1 Installation

Set the following environment variables for the userid that will run LEI:

- ▶ Set the environment variable LANG to your proper locale. If this is not set, the default locale of “C” will be used:

```
LANG=C
```

- ▶ Set the environment variable LOTUS to the canonical Lotus directory /opt/lotus. The canonical Lotus directory is the directory where all Lotus software is installed:

```
LOTUS=/opt/lotus
```

- ▶ Set the environment variable Notes_ExecDirectory to specify the Notes executable directory:

```
Notes_ExecDirectory=/opt/lotus/notes/latest/sunspa
```

- ▶ Set the environment variable PATH to include the following directories:

- Notes Resource directory:

```
$LOTUS/notes/latest/sunspa/res/$LANG
```

- Lotus executable directory:

```
$Notes_ExecDirectory
```

- LEI directory (for example /export/home/lei)

- Notes data directory (so your Notes.ini file can be found, for example: /export/notesdata).

- The PATH environment variable at the end should be configured as follows:

```
PATH=$PATH:/opt/lotus/notes/latest/sunspa/res/$LANG:/export/home/lei:/opt/lotus/notes/latest/sunspa:/opt/lotus/lei:/export/notesdata
```

- ▶ Set the environment variable LD_LIBRARY_PATH, which is used to locate shared libraries, to include the following directories:

- LEI directory (for example /opt/lotus/lei)

- Notes executable directory `$Notes_ExecDirectory`
- Any other product library directories you may require (for example: `$ORACLE/lib`, `$SYBASE/lib`, `$ODBC_HOME/lib`).

For example:

```
LD_LIBRARY_PATH=/opt/lotus/lei:/opt/lotus/notes/latest/sunspa:$ORACLE/lib
```

You must be logged in as the owner of the R5 Domino installation when launching LEI setup. Prior to completion of the installation process, you will be prompted to enter the root password for your system. Have this handy before starting the installation.

If you reach the point of the installation where you are prompted for the root password and you cannot provide it, press Ctrl-c to terminate the setup process.

After acquiring the correct password, run **setup finishinstall** from the LEI program directory to complete the installation process. Do *not* run **setup finishinstall** from the distribution media.

LEI requires a minimum of 35 MB of file system space. If you are not sure, verify sufficient space exists for your intended LEI target directory files before running setup (use **df -k <intended target dir>**). Setup does not check for available space.

Be sure the Domino server that will host the Administrator database for this LEI server is up and running, even if it is located on the machine to which you are installing.

If the installation uses the same UNIX user ID to run the Notes server and LEI, running **leiclean** will terminate the Notes server. In addition to cleaning up the LEI processes, it also terminates the Notes server processes.

Restart the Domino server after setup completes, since setup may update files used by DECS.

10.2.2 Running LEI

The LEI server must be running in order to execute LEI Activities. Start the LEI server by executing the LEI program **lei** from the LEI user:

```
# lei
```

The LEI server window appears, displaying the LEI server commands.

When started, the LEI server connects to the LEI Administrator database and immediately runs any overdue Activities in the Administrator database.

You can also run the LEI server as a Domino add-in task. Issue the following command from the Domino server console:

```
> load lei
```

To manually shut down LEI when running as an add-in task, enter the command **tell lei quit** from the Domino server console, as a normal Domino task.

To send LEI console commands, submit the command **tell lei <command>** to the Domino server console.

For example, the command **tell lei c 1** will have the same effect as entering **c 1** at the standard LEI server console, which is to attempt to close Activity #1.

10.3 ODBC drivers

ODBC is a standard to which many database vendors have written drivers that allow you to read, write, and query information from their databases from other, external applications.

In our test lab we installed the ODBC driver from Intersolve, using the evaluation copy of the Connect ODBC 3.6 server. We did a default installation, with all the software installed in the /opt/odbc directory.

10.3.1 Configuration

In order for Domino to connect to a back-end data source (DB2, Oracle, Sybase, and so forth) using the ODBC interface, ODBC must be configured correctly.

Lotus does not supply ODBC drivers or ODBC administrators; they must be supplied by the back-end data source vendor or a third-party ODBC driver vendor. Follow the instructions provided with the ODBC driver/administrator and confirm that ODBC connectivity has been configured correctly prior to trying to connect via Domino.

To configure a data source, you must edit the system information file, a plain text file that is normally located in the user's \$HOME directory and is usually called .odbc.ini.

When Domino tries to connect to a data source, it makes a request to the ODBC administrator and passes the data source name (DSN), username and password parameters. The ODBC administrator then looks in the .odbc.ini file for the section that matches the DSN parameter. This section contains the information needed to connect to the back-end data source.

The Domino server runs under a UNIX user. In order for Domino to be able to use ODBC, this user must be able to see the .odbc.ini file.

The directory where this file resides needs to be added to the Domino user's PATH environment variable. Users should also check to make sure that the UNIX user running Domino can only see one instance of the .odbc.ini file.

There must be an [ODBC] section in the system information file that includes the InstallDir keyword. The value of this keyword must be the path to the directory under which the /lib and /messages directories are contained. For example, if you choose the default install directory, then the following line must be in the [ODBC] section:

```
InstallDir=/opt/odbc
```

Following is an example of an .odbc.ini file for DB2 and Oracle:

```
[ODBC Data Sources]
Oracle7=Sample Oracle dsn
DB2=Sample DB2 dsn
[Oracle7]
Driver=/opt/odbc/lib/ivor7 xx.so
Description=Oracle7
ServerName=oraclehost
LogonID=odbc01
Password=odbc01
[DB2]
Driver=/opt/odbc/lib/ivdb2 xx.so
Description=DB2
Database=ODBC
[ODBC]
Trace=0
TraceFile=odbctrace.out
TraceDll=/opt/odbc/lib/odbcdrac.so
InstallDir=/opt/odbc
```

Note: In this example, xx represents the driver number.

Another environment variable for the Domino user is the LD_LIBRARY_PATH variable. This variable contains the path for the Shared Libraries. In our case this variable must contain the path to the ODBC Shared Libraries, by default in /opt/odbc/lib.

Note: If you use the LD_LIBRARY_PATH you must include the Domino program directory that contains the Domino Shared Libraries, /opt/lotus/notes/latest/sunspa.

10.4 Troubleshooting

You can use the following checklist to troubleshoot common problems on DECS and LEI connecting to an Oracle server:

- ▶ Does the environment variable ORACLE_HOME point to the Oracle home, usually /home/oracle?
- ▶ Does PATH include the Oracle bin directory, usually /home/oracle/bin?
- ▶ Does the LD_LIBRARY_PATH include the Oracle lib directory, usually /home/oracle/lib?
- ▶ Does the profile for the User ID that starts Domino include these variables?
- ▶ Does that User ID have permission to access and execute the files in those Oracle directories?

10.4.1 Test the Oracle environment

Using the Oracle client software (sqlplus), and without any Domino or LEI involvement, confirm that a connection to the Oracle server can be made and that a simple SQL statement can be executed.

For example:

```
cd $ORACLE_HOME/bin
sqlplus <username>/<password>@<host string>
select * from emp;
```

This will show all the records from the emp Oracle table.

10.4.2 Using the LCTEST tool

You can verify the connectivity with your remote database system by using the LCTEST program supplied with Domino.

You can find the program in the Domino program directory /opt/lotus/notes/latest/sunspa.

Using the Domino user account, issue the command:

```
# lctest
```

You can use the LCTEST tool to test connectivity with the following databases:

- ▶ Lotus Notes
- ▶ Oracle server
- ▶ ODBC
- ▶ Sybase server

- ▶ EDA/SQL
- ▶ DB/2

After you choose the system that you want to test, you will be prompted for your Username, Password, and Hoststring.

The LCTEST must come back successfully for DECS and LEI to operate properly. If it does not, you must resolve any issues with your back-end client before Domino can continue troubleshooting the issue.

10.4.3 Using the CONTEST tool

You can verify the connector validity with your back-end system using the CONTEST program supplied with Domino.

You can find the program in the LEI program directory, by default in /opt/lotus/lei.

The CONTEST tool is an additional testing program, similar in concept to the connector-specific test program LCTEST.

CONTEST must be run with a running LEI server. It attempts to connect by using connections defined in the currently running LEI server's Administrator database.

CONTEST tests the ability to make a connection through the information found in the Connection document.

Log in with the Domino user account to use the command. The syntax for using CONTEST is as follows:

```
# contest [-p] <connector1> <connector2>...<connectorn>
```

The -p option displays the connector properties.

For example, this command will test an Oracle connection as defined in LEI:

```
# contest -p <Oracle connection name>
```

Tip: Typing **contest** with no input parameters results in Help information being displayed.

10.4.4 Checking the shared library

Use the Solaris **ldd** command to determine whether the shared library can be loaded. The **ldd** command lists dynamic dependencies of executable files or shared objects.

For example, to check the connection with Oracle, go into the Domino program directory /opt/lotus/notes/latest/sunspa and issue the command:

```
$ ldd oracle.lcx
liblcapi.so =>      ./liblcapi.so
libclntsh.so.1.0 => (file not found)
libdl.so.1 =>       /usr/lib/libdl.so.1
libm.so.1 =>        /usr/lib/libm.so.1
libc.so.1 =>        /usr/lib/libc.so.1
libthread.so.1 =>   /usr/lib/libthread.so.1
libsocket.so.1 =>   /usr/lib/libsocket.so.1
libC.so.5 =>        /usr/lib/libC.so.5
libnsl.so.1 =>      /usr/lib/libnsl.so.1
libw.so.1 =>        /usr/lib/libw.so.1
libmp.so.2 =>       /usr/lib/libmp.so.2
```

In this example the Oracle client library libclntsh.so.1.0 is not found.

If the library cannot be loaded successfully, this will identify where the dependencies are failing. Confirm that these dependencies exist.

10.4.5 leiclean

A shell script is provided with LEI installations on UNIX platforms to clean up LEI resources following an abnormal termination of LEI. The shell script, named leiclean, is located in the LEI binary directory (/opt/lotus/lei by default).

It should be used with caution. It kills all LEI processes owned by the UNIX user who executes the shell script and frees all shared memory and semaphores currently in use by that user.

Use this script only after an abnormal termination of LEI. In addition to terminating all LEI processes, the removal of shared memory and semaphores could affect any other programs that are currently running under the same UNIX userid. In particular, this includes the Domino server and any other Notes API programs that the same UNIX userid is currently executing.

If you execute LEI and the Domino server under the same UNIX userid (as you must to use RealTime), an abnormal termination by either process may halt the other.

10.5 Summary

In this chapter we discussed how you can install, test, and troubleshoot the enterprise integration features of your Domino server.



Backup strategy for Domino R5 on Solaris

Backup strategies for Domino have been in place since early implementations. In this chapter we discuss the different mechanisms for backing up Domino in a Solaris environment and the new features available in Domino R5 for backup.

In Domino R5 a new application program interface (API) for backup and recovery was introduced. In this chapter only vendor solutions which use this API with its new features will be described. Backup and recovery procedures prior to Lotus Domino R5 are not covered in this chapter.

11.1 Backup strategy

A typical backup is performed by using one backup server to back up all your Domino servers through your LAN or a dedicated backup network.

Figure 11-1 presents a diagram of a typical backup in Domino.

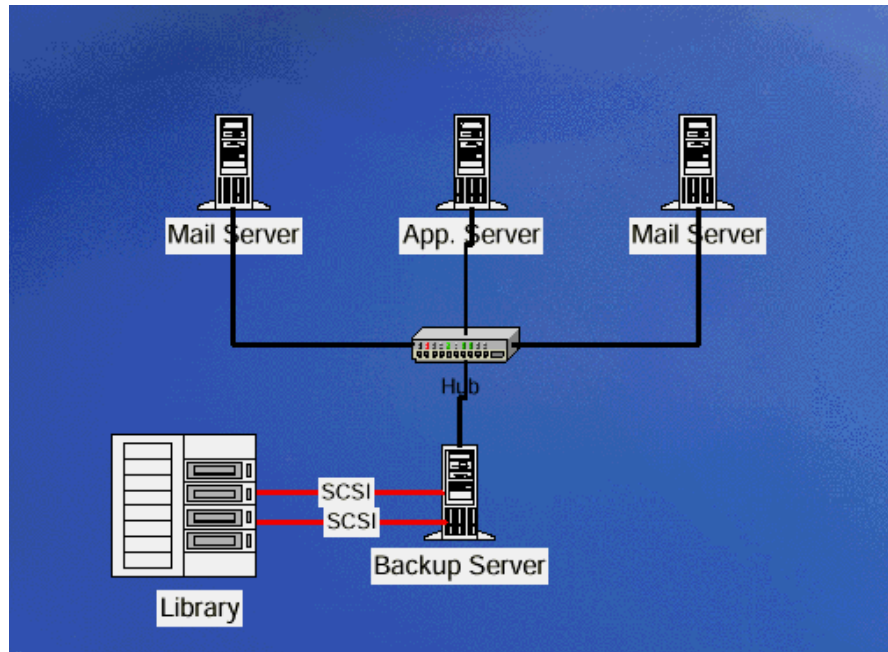


Figure 11-1 Domino backup scenario

In the diagram we show three servers that are backed up to one backup server using a tape library unit to enable backup of multiple servers. You can also back up individual servers by attaching a single tape unit to a Solaris system and backing up the servers individually. Once you have set up your backup scenario, you will need to decide on the method that you will use to back up your data. The significant challenge with backing up Domino databases is handling open files. Because this situation is common with just about all database systems, most major backup systems have solutions to this problem.

There are two basic methods to back up Domino:

- **Offline backup:** Shut down the server, back up your files, then restart the server. This is the most reliable and inexpensive type of backup procedure. This accomplishes all the necessary file integrity results with very little cost. The downside to this is that it cannot be done on critical systems that require

24/7 operation. One way to have Domino still in production during offline backup is to have a Domino cluster where one server is dedicated for backup and thus shut down during the backup procedure. This can be done during non-peak hours of operation. Be aware that you lose all the cluster benefits for this server, like failover or load-balancing, during the backup time. After the server resumes its place in the cluster, a replication has to be set to synchronize the data that was created on the other cluster members during the backup period. (See Chapter 5, “Domino advanced services” on page 111 for details on synchronizing data across a cluster.)

- ▶ **Online backup:** Online backup provides a way to back up your data and still have your system in production. This option becomes more and more important with the requirement of 24/7 operation becoming more common. There are different options to perform an online backup. We recommend that you utilize the features provided by the backup/recovery APIs in Domino R5. You can then use third party tools or write your own backup software. Implementing these two methods are the topic of the rest of this chapter.

11.1.1 Using Solaris backup utilities

There are a number of utilities provided with your Solaris OS that can be used to perform backups. An overview of these utilities follows.

- ▶ **CPIO** - This utility is a UNIX system backup procedure that has been in existence since the early implementations of the UNIX operating system. Files can be backed up and restored from disk or tape.
- ▶ **TAR** - This utility is a UNIX system file archive procedure that has gained popularity on all the UNIX platforms.
- ▶ **UFSDUMP/ UFSRESTORE** - This utility is used specifically for backing up file systems on the Solaris OS. There are many options that can be used with this procedure, and it can be a quick way to back up your Domino servers without purchasing any additional software.
- ▶ **DD** - This is one of the original “dump” utilities that is used on a UNIX system to write files to disk and tape. It is not very user friendly, but it can be useful in experienced hands.

Consult your Solaris product manuals or online documentation for detailed procedures for backing up files and directories using these utilities.

11.1.2 Backup device options

Another consideration when backing up files on a Solaris system is which of the available backup devices to use when you back up on the same machine and are not using a backup network.

Almost all devices configured on a Solaris system are in the /dev directory and are linked to special files in the /devices directory. When you install the Solaris system, the special files are created based on the devices that are attached to your system. If you add additional devices, such as a tape drive or tape library, you will need to configure the drive in the Solaris OS before you can use it with any software that you have installed.

The standard backup devices are disk, tape, and floppy. An example of the type of name associated with each of the devices follows:

- ▶ Floppy - /dev/rdiskette
- ▶ Tape - /dev/rmt/0
- ▶ Disk - /dev/dsk/c0t0d0s0

When you attach a tape drive to your Solaris system, you should use the following steps to ensure that the device is recognized by the Solaris OS.

1. If the tape drive is a SCSI device, set the SCSI ID before connecting the device.
2. Connect the tape device with the appropriate cable that came with your tape drive, or consult the manufacturer for the correct cable.
3. Reboot your system with the “-r” option for the reconfiguration of the devices to be accomplished.

Tip: The command is **reboot - -r**

4. Change to the /dev/rmt directory and type the following command: **ls -la**

This command will produce a list of device names that were created for your tape drive. The characters following the drive number have special meaning to the device and should be used as instructed by the software or the Solaris utility you are using.

For further instructions on setting up the device to back up your Domino data using UNIX utilities, consult the Solaris product manuals or online documentation. For instructions for backup with a separately purchased third-party product, consult the documentation provided by the manufacturer.

11.2 Backup management

In this section we discuss the management issues related to backing up files, such as why you still need backups even if you are replicating your databases, how to establish backup cycles, and how to implement incremental backups with the transaction logging feature enabled in Domino R5.

11.2.1 Backup versus replication

Your Domino implementation may include clustering of your Domino servers so you can replicate your databases to another system or disk. What we want to point out in this section is that replication does not replace the need to have reliable backups of your databases.

It is true that in the event of disk failure or disaster recovery a replica of a database is a quick way to recover the information that was lost, but there are other times when a database may need to be recovered from a previous day or week. Listed below are some cases when you would need a backup of data.

- ▶ Information that was in a database was changed and this was discovered at a later date. Replication has already overwritten the changed information on the cluster pair.
- ▶ A database has become corrupted on the server; this was not discovered prior to corruption replicating to the cluster or other replicas.
- ▶ An Adminp request was issued and approved to perform deletion of databases through your servers. This was discovered but could not be stopped prior to user databases being deleted.
- ▶ A user has inadvertently deleted all mail in their database and did not inform the administrator in time to stop replication.

These are just a few examples of why a reliable backup to your databases is an important part of your Domino implementation planning.

11.2.2 Backup cycles

When planning for your backup, it is a good idea to develop a backup cycle that will work for your organization. You should consider the following issues when determining a good backup cycle for your office:

- ▶ Budget allotment for tapes and life cycle of tape usage
- ▶ Company policy on mail retention and archiving
- ▶ Amount of data to be backed up per server
- ▶ Time available for backup

11.2.3 Incremental backups versus full backups

With the release of Domino R5 and the transaction logging feature, you have the capability of doing incremental backups to the transaction logs. You need to have at least one full backup of your databases. For recovery reasons it is better to do full backups on a frequent basis (for example, once a week) and between the full backups do an incremental backup to the transaction logs.

In this section we discuss the pros and cons to consider when choosing to use either an incremental backup or a full backup solution. Transaction logging is an essential part of the Domino server since it provides performance optimizations as well as the log used for backup. We discuss the ways in which transaction log backups and full backups can be used; you must determine which way suits your particular situation. You should review the documentation on transaction logging found in the Administration Guide for R5 to get a full understanding of the operation of transaction logs.

Domino R5 introduced transaction logging for recovery. With this feature enabled, the system captures database changes and writes them to the transaction log. Then, if a system or media failure occurs, you can use the transaction log and a third-party backup utility to recover your databases.

A single transaction is a series of changes made to a database on a server. An example of a transaction might include opening a new document, adding text, and saving the document.

Transaction logging is only available for databases with Domino R5 on disk structure (ODS) file format. You should keep this in mind when planning your backup strategies.

When you enable transaction logging you must select the type of logging that you want to occur. You can choose circular logging (which is the default) or archive logging.

- *Circular Logging* continuously re-uses the log files and overwrites old transactions. You are limited to restoring only the transactions stored in the transaction logs. If this implementation is selected, full nightly backups are required. The maximum size of circular transaction logs is 4 GB in total. If the overall size of your transactions exceeds this size, the oldest logs will be overwritten. Be aware that you can lose data if the backup procedure is not scheduled appropriately, meaning that the logs are full and overwritten before you backed them up! This needs to be monitored—especially in heavy loaded environments. Each backup takes longer to perform, but the restore process is more efficient because only the most recent (or other appropriate) full backups need to be restored. You cannot archive the transaction log if circular logging is used. Therefore, if you lose both the database and the recovery

log, you will only be able to recover the database at its state at the last backup.

- ▶ *Archive Logging* does not re-use the log files until they are archived. A log file can be archived when it is inactive, which means that it does not contain any transactions necessary for a restart recovery. Use a third-party utility to copy and archive the existing logs. The archive log files will be created incrementally according to a set schedule. With this implementation, incremental backups of the transaction logs can be accomplished daily, with full backups run, for instance, once a week or when a situation occurs that changes the database instance ID. A full backup once a week reduces the number of transaction log extents to be processed during a restore. It also reduces the number of transaction logs and therefore the disk space required to store them. The disk you dedicated for transaction logging has to be large enough because it can cause severe trouble if the server runs out of disk to write transactions into the logs. This has to be monitored!

The next consideration for setting up transaction logging is the way the database instance IDs are created and maintained.

When you enable transaction logging, Domino assigns a database instance ID (DBIID) to each Domino Release 5 database. When Domino records a transaction in the log, it includes the DBIID. During recovery, Domino uses the DBIID to match transactions to databases. Some database maintenance activities, such as compaction with options, cause Domino to assign a new DBIID to a database. From that point forward, all new transactions recorded in the log use the new DBIID. Since the previous transactions have a different DBIID, you would not be able to restore any data from the old logs. When these situations occur you will need to perform a full backup of your databases.

Note: When the Domino server is installed, compaction of databases is performed daily by default. Change the compact task to a weekly housekeeping procedure and create a full backup of your databases after the compaction is complete.

Following are some of the cases when Domino assigns a new DBIID to the transaction logs, requiring a new full backup:

- ▶ Transaction logging is enabled for the first time
- ▶ A Compact server task with options is run
- ▶ Fixup is run on any databases that were corrupted
- ▶ The log path or maximum log size is changed. A Domino R5 database is moved from one logged server to another logged server, or from an unlogged server to a logged server.

As you can see from this list, there are a number of considerations for implementing incremental backups at your location. Close analyses of all these variables should be accomplished before a final decision on your method of backup is made. Whether you decide to perform incremental or full backups, test your procedures regularly to ensure the accuracy of the data to be restored. In most cases you will have a mixed environment with full and incremental backups running each day.

11.2.4 Backup using Lotus C API for Domino R5

In this section we describe the Lotus C API for Domino and Notes, which can be used to write your own software to back up and restore the transaction logs in Domino R5.

If transaction logging is enabled on the server, all Domino R5 format databases in the server data path are logged by default. The Domino administrator can disable logging for a particular database. Earlier database versions are not supported by transaction logging. All logged transactions go into a single *transaction log*, consisting of one or more files or extents. Transaction logging may be of either circular style or archive style. Transaction logging must be enabled in order to implement the recovery of databases via the API's backup and recovery functionality.

A full technical description of the functions of the Lotus C API for Domino and a sample c-script are in Appendix A.

The API allows the backup products to perform the following functions:

- ▶ Online backup of R5 databases
- ▶ Maintain multiple backup versions of R5 databases
- ▶ Archive transaction log extents (if archival logging is used)
- ▶ Restore any version of an R5 database and apply changes since the backup from the transaction log
- ▶ Restore R5 databases to a specific point in time
- ▶ Restore one or more archived transaction logs
- ▶ Expire database backups automatically based on version limit and retention period
- ▶ Archive inactive transaction log extents when they are no longer needed for restore
- ▶ Automate scheduled backups

(See the redbook *Backing Up Lotus Domino R5 Using Tivoli Storage Management*, SG24-5247, for more details.)

11.2.5 Considerations for backup software

When you select third-party software, there are some features that relate to your Domino server that should be considered. Your evaluation of the software should determine whether it provides the following capabilities:

- ▶ On-line full and incremental backup of Notes databases.
- ▶ Off-line full and incremental backup of Notes databases.
- ▶ Selective network port addressing for backup across a LAN. This is valuable if you have installed a private network for your clustering. You can back up your servers without using the bandwidth necessary for the Domino server functions.
- ▶ Automatic discovery of new Notes databases.
- ▶ Automatic recognition of DBIID change to select full instead of incremental backup for transaction logging.
- ▶ Software determines which transaction logs are aged (obsolete) and informs you or deletes logs.
- ▶ On-line recovery of Notes entire database.
- ▶ Off-line recovery of single or multiple Notes databases.
- ▶ Automated backup scheduling for Domino server.
- ▶ Automated backup scheduling by Domino databases.
- ▶ Centralized administration of distributed Notes environment.

11.3 Vendor solutions

There are a number of third-party software companies that offer backup software for the Domino server on the Solaris platform. Some vendors have products that support the Domino R5 C API for backing up the transaction logs. Table 11-1 on page 290 shows the existing vendor solutions that use the R5 backup/restore APIs for all platforms. These solutions only apply Archive Transactional Logging but not Circular Transactional Logging.

Note: A supported backup utility backs up the transactional log files (.TXN), not the database files (.NSF).

Table 11-1 Vendor solutions for backup/restore using R5 APIs

Company	Product Name	NT	WIN2K	OS/390	AS/400	AIX	Solaris	Linux
Legato Systems Inc. http://www.legato.com/	Networker Module for Lotus	X	X			X	X	Beta
Tivoli http://www.tivoli.com	Tivoli Data Protection for Lotus Domino	X	X	X	X	X	X	
Veritas Software Corp. http://www.veritas.com	Netbackup 3.4	X	X			X	X	
Commvault http://www.commvault.com	Commvault Galaxy	X	X					
EMC http://www.emc.com/	EMC Data Manager for Lotus Notes Domino R5 Servers	X	X					
Computer Associates http://www.ca.com	ARCserve 2000 Lotus Notes Agent Version 7.0, Build 268.	X	X					

As you can see, the vendors with solution for Domino R5 on Sun Solaris are (in alphabetical order):

- Legato Systems, Inc
- Tivoli
- Veritas Software Corp.

There are other third-party products that can be used to back up Domino that are not listed here. If you are considering purchasing such software, be sure to get an evaluation copy and test the software to verify that it can support the backup strategy for your location. Remember that if you want to back up Domino R5 databases that have transaction logging turned on, you will need software that was written specifically for the R5 C API.

11.3.1 Legato NetWorker Module for Lotus

At the time of this writing, NetWorker Module for Lotus Notes Release 2.1 is the newest version. It is installed on the Lotus Domino Server and has the capability to search out all the Notes databases on that server, using one of three user-specified search methods: the explicit filename, the standard notes directory, and a search of the whole machine. It reads the database files, formats them into a NetWorker savestream using XOpen's Backup Services API (XBSA),

and passes the data to the NetWorker server. The NetWorker server may be either on the same machine, or on another machine on the network. The NetWorker Module can also be installed on the Notes Client to back up any databases that reside on that machine.

Performance in NetWorker is achieved with parallelism that allows many databases to be backed up concurrently, thus dramatically reducing the amount of time required to perform a backup. This works via a parallelism value, which can be set by the user. This value is the maximum amount of streams that NetWorker will use when passing data to the NetWorker server. NetWorker then automatically determines the optimal parallelism for the database configuration, to minimize contention on the disk drives. The result is that there will be a parallel stream for each disk drive containing Notes databases. On restore, in order to reduce tape device contention, parallelism is set to the number of tape devices being used for the restore.

An overview of the features in Legato NetWorker Module includes the following:

- ▶ On-line, non-disruptive backups
- ▶ Full or incremental backups
- ▶ Document-level backup and restore (phase2)
- ▶ Point-in-time restore and directed (to another directory) restore
- ▶ Autochanger support
- ▶ Media management (tape tracking, labeling, and bar code support)
- ▶ User notification by email and log files
- ▶ Graphical backup and recovery interface (NT only)
- ▶ Graphical scheduling interface
- ▶ Seamless integration of Notes backup with file system backup for enterprise-wide storage management
- ▶ Local or remote backup and restore
- ▶ Optional data compression and encryption

Installation requirements

To install the NetWorker Module for Lotus software, your database server/client must be equipped with the following:

- ▶ Solaris operating system, version 7 or 8.
- ▶ Lotus Notes/Domino Server, versions 5.04 or 5.06.
- ▶ NetWorker client software, version 5.1 or later.

- ▶ The NetWorker server requires either NetWorker for Windows server software or NetWorker for UNIX server software, version 5.1 or later.

Files installed during installation

The NetWorker Module software installs the following files:

- ▶ NetWorker Module for Lotus binaries and programs in /usr/sbin:
 - nsrnotesv, the backup command
 - nsrnotesrc, the recover command
 - nsrdocrc, the document-level recovery command
 - nsrnote, the backup script that NetWorker software calls during a scheduled backup
 - nsrnml_remrecov, a script that allows the NetWorker Module software to perform a remote recovery from the NetWorker User for Lotus software on Windows
- ▶ Man pages in /usr/man/man1m:
 - nsrnotesv.1m
 - nsrnotesrc.1m
 - nsrdocrc.1m

Installing the NetWorker Module for Lotus on Solaris

To install the NetWorker Module for Lotus software on a Solaris computer running Lotus Notes/Domino server or client:

1. Begin the installation process using the **pkgadd** command:

```
# pkgadd -d nml.pkg
```

or

```
# pkgadd -d nml_solaris.pkg
```

2. When the script prompts “Select package(s) you wish to process:” enter [1] to install the NetWorker for Solaris (Backup/Recover) NML package. The NetWorker Module for Lotus software must be installed into the same directory as NetWorker software.
3. If the libnotes.so file is not present in the default location /opt/lotus/notes/latest/sunspa, you will be prompted for the correct path to libnotes.so.
4. When prompted for Lotus information, enter the user name and group name that you created when you installed Lotus Notes/Domino. If you used, for example, “notes” for the user and group names, you would enter:

Please enter Lotus user name: **notes**

Please enter Lotus group name: **notes**

5. Create \$Notes_ExecDirectory and \$LD_LIBRARY_PATH environment variables and set them to the location of the libnotes.so file, which is typically /opt/lotus/notes/latest/sunspa.
6. Set the \$PATH environment variable to the location of the following items:
 - NetWorker Module for Lotus - typically /usr/sbin
 - Lotus Notes/Domino binaries - typically /opt/lotus/bin
 - Notes Exec Directory - typically /opt/lotus/notes/latest/sunspa
 - Lotus Resource Directory - typically /opt/lotus/notes/latest/sunspa/res/C
 - Lotus Notes/Domino data directory - defined by user during installation

Enable and register NetWorker Module for Lotus

To enable NetWorker Module for Lotus software:

1. Start the NetWorker administration program and connect to the NetWorker server.
2. Go to the Registration window and select Create.
3. Enter the enabler code in the Enabler code field.

The enabler code depends upon whether you have purchased the NetWorker Module software, or whether you are evaluating it.

Troubleshooting

If you have trouble installing and using this program, consider the following as first steps in troubleshooting:

- ▶ Use the -P1 option. The parent process does most of the work, and involves a lower number of processes.
- ▶ Get the output generated by running the command with the -D9 option. This will create maximum debug messages.
- ▶ Examine the batch file/script used for starting scheduled backups. (nsrnote) Check whether all environment variables and arguments are set properly.
- ▶ Check for correct User/Group IDs.
 - Save has to run as root. The binary should have the setuid bit set for user and group. Owner of the binary should be the same as that running domino server.
 - Recover has to be run as the Domino user.
- ▶ Use the -Z option when recovering DB in different locations, when the DB exists in the Domino Data directory.

- ▶ If you have a problem recovering transaction logs, the `-l<n>` option can be used to recover transaction logs independent of databases.

11.3.2 Tivoli Data Protection for Lotus Domino

The information in this section was excerpted from the IBM Redbook *Backing Up Lotus Domino R5 Using Tivoli Storage Management*, SG24-5247.

Tivoli Data Protection for the Lotus Domino application client provides an integrated solution for performing full backup and restore operations on Lotus Domino R5 databases and database templates. It is a client application that provides full backup of online databases and restore of full databases to the original or different location. Tivoli Data Protection for Lotus Domino also archives the transaction log extents of a Domino server and retrieves the appropriate transaction log extents for the recovery of databases if archive transaction logging is enabled on the Domino server. Tivoli Data Protection for Lotus Domino is not intended as a substitute for the standard Tivoli Storage Manager backup/archive client. Tivoli Data Protection for Domino cannot be used to back up or restore any non-database data, such as Notes ID files, or Notes.ini, or any other system configuration files. Those files need to be backed up by the Tivoli Storage Manager backup/archive client. Therefore, the two client types work together to provide full data protection for your Notes environment.

The Tivoli Data Protection for Lotus Domino application client and the Tivoli Storage Manager backup/archive client can run simultaneously on the same Domino server; however, they are totally separate clients as far as the Tivoli Storage Manager server is concerned.

Tivoli Data Protection for Lotus Domino provides the following actions and operations:

- ▶ Perform full backup of online databases (.nsf) and templates (.ntf).
- ▶ Perform conditional full backup, incremental backup of entire databases.
- ▶ Archive of transaction log extents, if archive transaction logging is enabled on Domino server.
- ▶ Restore any backup version of a database and apply changes since the last backup from the transaction log.
- ▶ Restore a database to a specific point-in-time.
- ▶ Restore a database to another Domino server.
- ▶ Restore individual archived transaction logs.
- ▶ Expire database backups automatically based on version limit and retention period.

- ▶ Expire archived transaction log extents when no longer needed for the recovery of database backup versions.
- ▶ Queries of backed up databases, archived transaction log extents, and Tivoli Storage Manager server information.
- ▶ Queries of Domino databases and server information.
- ▶ Query and change of current values set in the preference file for Tivoli Data Protection for Lotus Domino.
- ▶ Change of Tivoli Data Protection client password.

The Tivoli Data Protection for Lotus Domino application client provides a command line interface for performing backups and restores. The application client commands are issued from a command prompt.

Unlike the Lotus Notes R4 API, Lotus Domino R5 uses an API specifically developed for backup and restore purposes. This API increases performance and reduces backup times.

Configuring Tivoli Data Protection for Lotus Domino

We recommend using unique node names, policy domains, and management class names across your configuration; that is, a combination Domino environment, Tivoli Storage Manager and Tivoli Data Protection for Lotus Domino. There may be references across the solution and the use of duplicate names may lead to serious problems and confusion.

Include/exclude lists

Tivoli Data Protection for Lotus Domino deals only with Domino databases and transaction log files (if archival logging is in effect on the Domino server). Other files that may exist on the server are not backed up by the Tivoli Data Protection for Domino application client, so they do not need to be excluded. However, if you want to limit the backups to a subset of the databases on your Domino server, the standard include/exclude syntax can be used.

Preference file

Another important file which will need modification and customization is the preferences file, `domdsm.cfg`. This file contains additional options used during backup and restore processing.

To update `domdsm.cfg`, you can either select Preferences from the Tivoli Data protection for Lotus Domino GUI Edit menu to define options and assign values in the preference file, or you can use the command line **`domdsmc set`** command to update the preferences file. You should not try to edit this file directly.

Tivoli Data Protection for Domino on UNIX option files

In a UNIX (including Solaris, AIX, and OS/390) environment, there are a number of configuration files required. These are:

- ▶ **dsm.opt**: Identifies the Tivoli Storage Manager server to contact and specifies backup and restore options. Also, this is called the client users options file. You can edit the dsm.opt file using your system editor.
- ▶ **dsm.opt.smp**: Sample options file that can be copied and modified if dsm.opt does not exist. You can edit the dsm.opt.smp file using your system editor. By default, Tivoli Data Protection for Lotus Domino will look for dsm.opt in the client installation directory.
- ▶ **dsm.sys**: Contains stanzas describing Tivoli Storage Manager servers to contact for services. These stanzas also specify communication methods, backup and restore options, and select scheduling options. Also, this is called the client system options file. If your system has a node name assigned to the Tivoli Storage Manager backup-archive client, we recommend you have a different node name and create a separate stanza in the dsm.sys system options file for the Tivoli Data Protection for Lotus Domino. Only the root or Tivoli Storage Manager authorized user can edit the dsm.sys system options file. This file is found in the Tivoli Storage Manager backup-archive client installation directory.
- ▶ **include/exclude list**: The location of this file is indicated by the INCLEXCL parameter in the dsm.sys file.

Domino server UNIX environment variables

The following environment variables are used to point to files and directories that Tivoli Data Protection for Lotus Domino uses:

- ▶ **DOMI_DIR**: This points to the directory where Tivoli Data Protection for Lotus Domino was installed. The default installation directory is /usr/tivoli/tsm/client/domino/.
- ▶ **DOMI_CONFIG**: This points to the Tivoli Data Protection for Lotus Domino preferences file. Specify this environment variable to change the default setting. The default is domdsm.cfg in the directory where Tivoli Data Protection for Lotus Domino is installed. The file name can include a fully-qualified path or a relative path. A relative path is the directory where Tivoli Data Protection for Lotus Domino is run.
- ▶ **DOMI_LOG**: This points to the directory where the Tivoli Data Protection for Lotus Domino log file (domdsm.log) will be stored. The default is the installation directory. Specify this environment variable to change the default setting.
- ▶ **DSMI_DIR**: This points to the directory where the Tivoli Storage Manager API is installed. This environment variable is required and there is no default.

- ▶ **DSMI_LOG:** This points to the directory where the Tivoli Storage Manager API error log file (dsierror.log) will be stored. Specify this environment variable to change the default setting. The default directory is the Tivoli Data Protection for Lotus Domino install directory.
- ▶ **DSMI_CONFIG:** This points to the Tivoli Storage Manager API options file name. Specify this environment variable to change the default setting. The default is dsm.opt file in the directory where Tivoli Data Protection for Lotus Domino is installed. The file name can include a fully-qualified path or a relative path. A relative path is relative to the current directory where Tivoli Data Protection for Lotus Domino is run.

The operating system path variable is also required:

- ▶ **PATH:** As of Domino server V5.0.2b and higher, the Domino resource directory must be added. The resource directory is the “res/\${LANG}” directory under the Domino executable directory. On Solaris, the directory is /opt/lotus/notes/latest/sunspa/res/\${LANG}.

Setting Bourne and Korn shell environment variables

For the Bourne or Korn shell, enter the environment variables in the .profile file of the user ID that runs the Domino server. DSMI_DIR and PATH are the only required environment variables.

Setting C shell environment variables

For the C shell, enter the environment variables in the .cshrc file of the user ID that runs the Domino server. DSMI_DIR and PATH are the only variables required.

Note: If your environment is not set up correctly, you may receive a Domino error when running Tivoli Data Protection for Domino. If the problem persists and the setup appears to be correct, there is a manual procedure to follow which may fix the problem. This procedure is documented in the README file contained in the application install package. You should carefully follow the steps, including the special considerations for partitioned servers. If this does not fix the problem, then contact Technical Support.

Precedence of Tivoli Data Protection option resolution in UNIX

Some options affecting the operation of Tivoli Data Protection for Domino can be specified in more than one way. The same option can derive from more than one configuration source. When this happens, the source with the highest priority takes precedence, in the sequence shown as follows:

- ▶ Tivoli Storage Manager backup-archive client system options file, dsm.sys.
- ▶ Tivoli Data Protection for Lotus Domino command line option, domdsmc.

- ▶ Tivoli Data Protection for Lotus Domino preferences file, domdsm.cfg.
- ▶ Tivoli Storage Manager backup-archive client user options file, dsm.opt.

Running the setup script on UNIX

After completing the installation and setting your environment variables, run the setup script, **domsetup**. The setup script assigns the group name, file owner, file permissions, and creates the symbolic link required by the Tivoli Data Protection for Lotus Domino executable. To run the domsetup script, perform the following steps:

1. Log on as the userid that was set up to run the Domino server.
2. Switch to the root userid by entering the **su root** command.
3. Change to the directory where Tivoli Data Protection for Lotus Domino is installed.
4. Enter **domsetup**

You can run the setup script without prompting by using the domsetup.dat file. Copy and edit the domsetup.dat file, follow the instructions within the file, and pass it as a command line argument to domsetup. If, for example, you create the domsetup.input file from the domsetup.dat file, use the following command:

```
domsetup domsetup.input
```

11.3.3 VERITAS NetBackup 3.4

NetBackup uses the Lotus Domino R5 backup/recovery APIs to perform on-line, non-disruptive backup of Domino databases, allow point-in-time recovery of Domino databases, and support the backup and automatic recovery of R5 transaction logs. NetBackup also supports backup of Domino clusters.

NetBackup may back up Domino databases as a server or as a client. Server backup implies the backup of data to direct-attached storage (disk or tape devices), or, in the case of Storage Area Networks (SAN), the backup of data through the backup server to tape device, away from the LAN (LAN-free backup), via tape drive virtualization software, such as NetBackup Shared Storage Option (SSO). Client backup implies backing up data to a VERITAS backup server.

Software requirements

The following software requirements are given for running VERITAS NetBackup for Lotus Notes 3.4:

- ▶ VERITAS NetBackup DataCenter 3.4 or higher, or VERITAS NetBackup BusinessServer 3.4 or higher and VERITAS NetBackup for Lotus Notes 3.4 or higher.

- ▶ Lotus Notes & Domino Server R5.0.3 or higher (support of Domino clustering in Lotus Domino 5.0.7 or higher), on the following platforms:
 - Sun Solaris 2.6, 7 (32-bit and 64-bit), and 8 (32-bit and 64-bit).

Installation

There are two ways of installing NetBackup. Both ways have in common that you need:

- ▶ NetBackup Server
 - ▶ NetBackup Client
 - ▶ NetBackup for Lotus Notes
1. On Domino servers to be configured as a NetBackup server, you must install both NetBackup server and client software prior to installing NetBackup for Lotus Notes.
 2. On Domino servers to be configured as a NetBackup client, you must install NetBackup client software prior to installing NetBackup for Lotus Notes. When server and client installation has been completed, you can install NetBackup for Lotus Notes.

NetBackup for Lotus Notes can be installed either remotely or locally. With remote installation, the administrator may load the software on the NetBackup Master Server and may then push NetBackup for Lotus Notes to the desired clients. Local installation loads and installs the agent only on the local machine. This section discusses local installation on UNIX. For additional details, refer to the VERITAS NetBackup for Lotus Notes System Administrator's Guide.

Local Installation

1. Log on as root user on the local machine.
2. If you're installing NetBackup for Lotus Notes on a NetBackup server, you must register a valid NetBackup for Lotus Notes key. To list or add NetBackup keys on UNIX, execute the following command:


```
install_path/netbackup/bin/admincmd/get_license_key
```

If you are installing NetBackup for Lotus Notes on a NetBackup client, proceed to step 3.
3. Insert the CD-ROM into the drive, and change the working directory to the CD-ROM directory.
4. Load and install NetBackup for Lotus Notes by executing the following install script:


```
./install
```
5. Upon executing the install script, the following prompt will appear:

Do you want to do a local installation? (y/n) [n]

6. Answer **y** to the prompt, select the NetBackup for Lotus Notes option, and enter **q** to quit selecting options. A prompt will appear asking if the list is correct
7. Answer **y** to the prompt. Immediately, the following actions will occur:
 - a. The version file, compressed tar file, and install_dbext script will be loaded to directory install_path/netbackup/dbext.
 - b. The install script will automatically execute the **install_dbext** script.
 - c. If **install_dbext** has successfully completed, there will be a version file in the directory install_path/netbackup/ext that contains the version of NetBackup for Lotus Notes on UNIX that was installed and an installation timestamp.

Configuration

This section provides a brief description of configuring a backup policy for NetBackup. For more detailed information, refer to the appropriate VERITAS product documentation.

VERITAS NetBackup offers different ways of creating and maintaining backup policies. Intuitive interfaces and even wizards (see Figure 11-2) assist the administrator in the formation of backup policy.

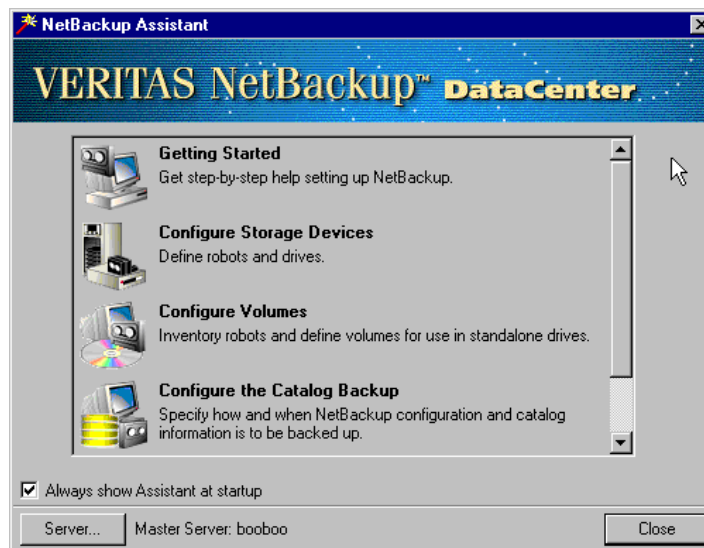


Figure 11-2 NetBackup wizards simplify administration

A VERITAS NetBackup backup policy may be divided into four primary areas: Attributes, Schedules, Files, and Clients. The details of these areas are as follows:

► **Attributes**

In Attributes, the administrator defines the kind of backup policy to be created. The administrator can also specify which storage unit (the logical representation of a physical device) will be used, or can let NetBackup automatically determine the storage unit based on availability. Furthermore, specific media volume pools can be designated by class.

► **Schedules**

In Schedules, automated backups can be configured to initiate within a designated time slot (backup window). The type of backup to be initiated is also specified in the Schedules window. For Domino, several backup options are available:

- Full Backup
- Differential incremental backup of unlogged databases or local databases, logged databases, and/or transaction logs
- Cumulative incremental backup of unlogged databases or local databases, logged databases, and/or transaction logs
- User backup

Each method performs slightly differently, so for additional details, refer to the VERITAS NetBackup for Lotus Notes System Administrator's Guide. Backup retention period and frequency of backup re-tries can also be specified in this window.

► **Files**

In Files, you can specify several file list directives in addition to paths to Lotus data. The following are a few of the directives available with NetBackup for Lotus Notes:

- `BACKUP_TRANSACTION_LOGS` will cause the backup of all transaction logs identified by Domino server as available for backup. For full and differential incremental backup types, transaction logs will be marked for recycling by Domino server.
- `NOTES_INI_PATH=` is utilized when using NetBackup to back up Domino partitioned servers. This directive identifies the location of the `NOTES.INI` file associated with the particular server partition that will be used to perform the backup.

► **Clients**

In Clients, the administrator designates which client machines will be backed up by a particular backup policy.

Troubleshooting

NetBackup offers a variety of standard reports that help administrators manage, monitor, verify, or troubleshoot their environments. Progress reports for both backup and restore operations are available, without having to manually parse transaction logs. As an option, NetBackup Advanced Reporter (NBAR) provides over 30 canned, graphical reports, delivered anywhere in the enterprise via Web browser. NBAR embeds hyperlinks within many reports, so that administrators may delve into greater detail without having to move to separate windows.

NetBackup also offers logging for cases where details reside outside of what standard or optional reporting can provide. For NetBackup for Lotus Notes, look for log information in the following directories :

install_path/Netbackup/logs/bpbkar (for backup operations)

install_path/Netbackup/logs/tar (for restore operations)

The administrator can also increase the level of logging by placing VERBOSE within the bp.conf file. For more information, refer to the NetBackup System Administrator's Guide or to the NetBackup Troubleshooting Guide.

11.4 Planning for successful backups

Once you have selected your backup software, you should take time to consider the issues that will affect a successful backup strategy. Among those considerations are the following:

- ▶ Will you be backing up across the network, and if so, will the bandwidth on your network be able to handle the load?
- ▶ Schedule your backups during non-peak hours. Your server performance will diminish if backup is running during high volume access periods.
- ▶ How many servers and databases will be backed up? Run test backups and estimate the amount of time you will need to complete your backups. Will you have enough time to complete them during scheduled hours?
- ▶ For servers with terabytes of data, architect your Domino directories as multiple file systems. Each of these can be backed up individually on a rotating basis, so that you will have more flexibility in the use of the backup tape capacity. If multiple file systems are not an option in your environment, use folders in your data directory.
- ▶ Consider the backup media you will be using, the cost of the media, and how much will be needed on an annual basis. Include this in your budget plans.
- ▶ Be sure to complete the weekly maintenance for your databases before running your weekly backups if you have implemented transaction logging

with incremental backups. Remember that the DBIID will change when compact with options and fixup are run on your databases.

- ▶ When considering your backup volume, be sure to take into account your transaction log size. The daily log size varies since it depends on the number of transactions in a database and the number of databases on a server. We recommend performing tests based on the user behavior in your company and monitoring the size of your transaction logs. Numbers usually range from 10% to 35% of the total database size for each server, but this is only an average from different environments and it might not adapt to your environment.
- ▶ Be sure to test the restore of your databases to ensure accurate data.
- ▶ Document your backup procedures and, if you are not the Domino administrator, inform the administrator of the effect transaction logging has on backup levels.
- ▶ If you use transaction logging with archive log enabled, be sure that the archive logs are recycled when not valid (when DBIID changes).
- ▶ Be sure you understand the throughput capabilities of any tape device you will use. If your backup is taking a long time to complete you may find that the throughput on your tape drive is the cause of your bottleneck.
- ▶ You can increase your throughput on a library by adding tape drives.
- ▶ If you are using single tape drives, consider having more than one.
- ▶ Software compression during backup can increase CPU utilization.
- ▶ When doing a restore with transaction logging:
 - Restore the database from the full backup first.
 - Next, restore all the transaction logs for the database.

11.4.1 Sample Schedule

This sample schedule for backups of transaction logs and logged databases has been taken from TN172508. It is common to follow this schedule in existing environments:

- ▶ Schedule daily incremental backups of the transaction log. Use the backup utility daily to back up the transaction log.
- ▶ Schedule archiving of transaction log files. If you use the archive logging style, use a third-party backup utility to schedule archiving of log files.
- ▶ Schedule weekly full database backups. Each week, it is recommended to run the Compact task with the option to reduce file size. Because this compaction style changes each database's DBIID, you should schedule compaction with a full database backup.

11.5 Summary

In this chapter we have discussed the performance of backups to your Domino server on a Solaris system. We described the types of backups that can be performed, from simple, inexpensive procedures to the use of third-party software. The configuration of devices that can be used for backup and the strategies to consider when planning a successful backup routine were also presented.



Diagnostics and troubleshooting

When problems arise, the pressure is on to get the server back up and running smoothly. The best time to start figuring out how to do troubleshooting or how to collect the appropriate data to enable customer support analysts to help you is not in the midst of a problem.

This chapter addresses three major topics:

- ▶ How to recognize when a problem is serious enough to place a call to Lotus Customer Support, and how to take the proper steps once the call is opened to use this resource most effectively. This means knowing what data will be useful to the analyst, how to collect it, and how to transmit it to Customer Support in a way that will get your problem looked at and resolved most quickly.
- ▶ Steps that can be taken on the client side to quickly isolate and fix some common problems.
- ▶ The importance of implementing recovery procedures in your company. This includes providing some sample scripts that can be tailored to your environment.

Note: The script provided in Chapter 3 will automatically use `nsd-kill` and bring the server down properly. The log file can be found in the directory defined for `nsd-logs` (e.g. `/lotus/nsd-logs/notes1`).

Note: The location of the Domino binary directory and the Domino data directory are configurable on install. All examples in this chapter are presented with the assumption that the default install path was chosen: /opt/lotus/ for the application binaries and /lotus/notes1/notesdata for the data directory. Change these paths as necessary to match your configuration.

12.1 Standard procedures

The following guidelines can be used for “panic” scenarios where recovery procedures and training are not yet in place. These steps can help reduce overall downtime and lead to a quicker resolution of the problem by providing data critical for initial analysis by Lotus Customer Support.

Step one is determining the nature of the problem. In most cases, this is one of the following occurrences:

- ▶ Crash (panic, fault, segmentation violation)
- ▶ Hang
- ▶ Performance issue

12.1.1 Crash

A crash is evidenced by the failure of the server to respond to requests; the following line appears on the server console:

```
Fatal Error signal = 0x0000000b PID/TID/K-TID = 1320/1208/3328  
Freezing all server threads ...
```

All the Domino processes are still running in a “freezing status;” this permits collection of all the information from the thread stack trace at the moment of the crash. There are also other messages that can indicate a crash—for example, anything that begins with the word “PANIC.”

When a crash occurs, capture a Notes System Diagnostic (NSD) by issuing the commands:

1. **cd /lotus/notes1/notesdata**
2. **/opt/lotus/bin/nsd**

Note: At this point, the nsd diagnostic script will scroll its output to the screen at the same time it is writing to a log file. The name of the log file will display when the script completes. The completion time could be anywhere from 5 to 20 minutes depending on the number of server processes or users.

3. **`/opt/lotus/bin/nsd -kill`**

This shuts down all the server tasks and cleans up memory.

4. **`/opt/lotus/bin/server > server.out`**

Or run your server startup script.

The NSD log file should be either tarred or zipped up, along with the last 500–2000 lines of the console log, and sent to the analyst assigned to the problem.

Tip: `tail -500 <consolefile.log> >server_console@date+time.log`

We discuss the NSD tool in detail later in this chapter.

12.1.2 Hang

A server hang is evidenced when the server stops responding and there is no indication of a problem in the console log: the server appears to have stopped functioning entirely and will no longer respond to client requests. This type of crash is generally more difficult to diagnose, and the data required to troubleshoot it is slightly different.

If the server is still not responding after a reasonable period of time has gone by (10 minutes or so), issue the following commands:

1. **`cd /lotus/notes1/notesdata`**

2. **`/opt/lotus/bin/nsd`**

3. **`/opt/lotus/bin/nsd`**

The name of the log file will display when the script completes. The completion time could be anywhere from 5 to 20 minutes depending on the number of server processes or users.

4. **`/opt/lotus/bin/nsd -kill`**

(to bring the server completely down)

5. **`/opt/lotus/bin/server > server.out`**

(or execute your server startup script)

The reason for the additional **NSD** is that **NSD** captures thread-level information which can provide more insight on whether this is actually a hang, or if the server is in a condition where it appears hung but is actually taking longer than usual to process requests.

The two created NSD files should be tarred or zipped, along with the console log, and sent to Lotus Support.

12.1.3 Performance problems

Performance problems typically are evidenced by a server which eventually gets the job done but is far slower than anticipated. Performance problems, crashes, and hangs are identical from an end-user perspective, since the client in every case is going to fail to establish a connection with the server.

Depending on the source of the problem, the type of information which leads to resolution can vary widely. As a general rule for initial problem analysis when no performance gathering tool is in place, you can treat the issue in the same way as a hang.

While the system is running in a degraded state, issue the following commands:

1. **cd /lotus/notes1/notesdata**

2. **/opt/lotus/bin/nsd**

Wait for five minutes after the NSD completes.

3. **/opt/lotus/bin/nsd**

4. **vmstat 1 100 > vmstat.out**

5. **iostat 1 100 > iostat.out**

6. **/opt/lotus/bin/nsd -kill**

(to bring the server completely down)

7. **/opt/lotus/bin/server > server.out**

(or execute your server startup script)

You should then tar the two NSDs, console log, vmstat.out and iostat.out, and contact Lotus Customer Support to open an incident report.

The **vmstat** Solaris command reports virtual memory statistics regarding process, virtual memory, disk, trap, and CPU activity. See the product documentation for more details.

Note: On multiprocessor systems use the **mpstat** command to report per-processor statistics. See the product documentation for more details.

The **iostat** utility iteratively reports terminal, disk, and tape I/O activity, as well as CPU utilization. The first line of output is for the entire time since boot, and each subsequent line is the average for the prior interval. See the product documentation for more details.

Important: If server performance is not improved by restarting the server, then you should disregard the last two steps when collecting data. The server may be experiencing a temporary performance degradation which may resolve itself after a period of time.

12.1.4 Packaging the files for support

Typically, the console log will have a file name such as server.out. Assuming the file name is server.out and also the NSDs are regularly moved out of the data directory, use the following commands to organize your files:

```
cd /lotus/notes1/notesdata
tar cvf files_for_support.tar console.log nsd_all*
```

This will tar up the files to one file called files_for_support.tar, which can then be FTPed (ensure you use binary mode to transfer the files) or e-mailed to Lotus Support.

Note: It is best to compress very large tar files using the standard UNIX command **compress**.

12.1.5 Transferring files to support

Once an incident is opened with Lotus Support, the support representative will ask for the relevant diagnostic information that you have collected.

Typically this is sent via e-mail or via FTP. When e-mailing files to Lotus, include in the CC: field of the e-mail the address incident_files@lotus.com and include the incident number as the subject line. This is a central repository which will allow any analyst to retrieve the files in case the owning analyst is unavailable to work the issue. You should not send any confidential data to this address, however.

To transfer the files via FTP prior to opening the incident you can send them to the following FTP site:

```
transfer.support.lotus.com
Login: anonymous
Password: your e-mail address (user@domain.com)
```

Inbound directory: /lotus/inbound/notes/

Files may be placed within the inbound directory. Please note that you will not have read access to the directory, so it will appear empty. If a file with the same name as yours already exists on the server you will get a “permission denied” message. Therefore, it is a good idea to name the file something unique, such as:

```
xxxarchivea.tar  
xxxarchiveb.tar etc...
```

where xxx is an incident number or company name. Always ensure you are using binary mode to transfer files or the file will become corrupt. On the command line FTP you can ensure this by typing **bin** before transferring the file.

12.1.6 Common mistakes seen by support

If you encounter problems when generating and sending files to the support office, consider whether you may have made any of the following common errors:

- ▶ Not using binary mode to transfer the files via FTP.
- ▶ Running an NSD with flags instead of an NSD with no flags (NSD -info).
- ▶ Running an **nsd -kill** before running NSD with no options.
- ▶ Incorrectly setting limits on the server.
- ▶ Running an older version of NSD. Be sure to request the latest version of the NSD tool from your Lotus support rep.

12.2 Crash, hang, and performance problem details

12.2.1 What is a crash?

A *crash* occurs when the application detects an unrecoverable fault. Crashes can manifest themselves in other ways as well, such as a hardware problem which may result in a system-wide crash.

There are a few ways to configure the Domino server to respond to a crash:

- ▶ Put all the threads into a spin so diagnostics can be gathered (default).
- ▶ Dump the core on the process which encountered the fatal error.

When the threads are put into a spin, the server stops dead in its tracks and an NSD may be taken to do a crash analysis in Domino.

The NSD will first attach to all the processes and obtain a trace from each thread. The trace will not make much sense to most administrators, since it will be a listing of all the functions the thread was processing prior to the crash.

Lotus Support can look at these traces and gain insight into how the application ran into the failure. This does not necessarily mean that the analyst will be able to come up with an exact cause for the crash because the same behavior can be seen for different types of crashes. Multiple occurrences of a crash and analysis of the diagnostics are sometimes required to sort out the cause from the symptoms.

For instance, if the failure occurs on an OS-level function that reads data from memory, the same trace could occur if the failure occurred due to memory corruption from an overwrite or from a faulty memory module.

12.2.2 What is a hang?

While a crash is a fault detected by the application, a *hang* is when the server appears to have stopped processing entirely but no messages are seen on the console indicating a problem.

Hangs can be thought of as either *recoverable* or *unrecoverable*.

A recoverable hang is one where after a period of time the server resumes servicing client requests, routing mail, responding to HTTP requests, and so forth. These are typically caused by an overloaded server or by insufficient resources, slower than required IO devices, and the like.

Unrecoverable hangs never clear themselves and a restart of the server is required to resume normal service. Unrecoverable hangs are typically caused by resource contention between one or more applications, processes, or threads.

Part of the difficulty in diagnosing hangs is that there is no “break point” indicating the nature of the problem, so a holistic approach is usually required to pinpoint the underlying problem.

When hangs occur, a good test to see if the Domino application is responding is to telnet to port 1352 on the Domino server.

If the server is not hung, you will get a message saying “Connection established.”

If it is hung, you will receive a “could not connect to remote host” or “connection refused” message.

12.2.3 Poor performance

Poor performance is typically caused by insufficient memory, slow disk response, insufficient CPU power, an incorrectly configured network, or an incorrectly configured Domino server.

As with hangs, a holistic approach is essential to quickly isolate the problem.

A **vmstat** will allow you to take a closer look at CPU utilization and an **iostat** will allow you to take a closer look at drive utilization.

There are also many third-party applications which can be used to monitor performance of the server processes and the server subsystems through graphs and charts. These can be invaluable for quickly identifying trends in performance.

Note: A good one is the proctool package from Solaris. Proctool is a system and process performance monitoring tool for Solaris 2.X. See <http://www.sunfreeware.com/> for more information about this free tool.

12.3 The NSD tool

Notes System Diagnostic (NSD) is a diagnostic script developed by Iris that gathers diagnostic information which can be used to troubleshoot problems and verify that the server is correctly configured. Version 3.4.6 is included with the Domino R5.08 server installation.

12.3.1 Running NSD

You must be in the data directory to run NSD. You can run NSD as the Notes user or as root. For most problems you will want to run the NSD as the Notes user.

You can use the **sudo** tool to permit the Notes user to have root privileges. See Chapter 7, “Security” on page 169 for more information about this tool.

Note: You can set the UNIX environment variable **NSD_LOGDIR** to point to the directory where you want your NSD file to be automatically saved.

Options for NSD

There are many different options that can be used with NSD to alter the type of information gathered. They are summarized in the following list.

Since NSD is constantly evolving and changing, new options may be added in the future. The **-help** option will show a complete list for the version of NSD you have installed.

-batch	run in batch mode; don't write to tty
-info	just report system info
-noinfo	don't report system info
-nolog	don't log output to log file
-ver*<i>sion</i>	just show version header
-ps	show process tree
-kill	kill all/user Notes processes and cleanup IPCs
-memcheck	run the Notes memory checker only
-nomemcheck	don't run the Notes memory checker by default
-lsof	run lsof only — list Notes open files
-nolsof	don't run lsof by default
-user <user_id>	operate only on Notes process run by 'user_id'
-exec_path <dir[:dir]*	add additional directories to the search path
-filter <log_file>	filter stack output of log_file
-help	show this help list
-help <option>	where option is any one of the above
-help gen*<i>eral</i>	general info about the script and how it works
-help lim*<i>itations</i>	general info on script limitations
-help update	list script version update info

Note: If the server restart time is a big issue for you, start the NSD using the PID of the crashing process as a parameter. In this way only the stack trace information of the crashing process will be collected. Be aware that some useful information can be missed when you choose this method.

The most notable options are:

- kill**
- info**
- nomemcheck**

Issuing **nsd -kill** will kill all Notes processes and clean up IPCS resources related to those processes.

Any time the server is not able to be shut down with a graceful quit from the console or a **server -q** from the command line, **nsd -kill** should be run to ensure that the environment is clean for a server restart.

The command **nsd -info** will skip attaching to the processes with a debugger and obtaining a trace. This is useful when you are only gathering system information and do not need any process-level information for diagnosis.

Lotus Support will often ask for the results from running **nsd -info** so they can do a initial assessment of the server environment.

Issuing **nsd -nomemcheck** will skip running memcheck against the application. Memcheck is a utility, developed by Iris, that obtains information on the current state of the Domino memory pools. Memcheck information may not be needed, and by using the **-nomemcheck** option you can reduce the total running time of the NSD script.

Note: The memcheck tool is not included with a standard installation, and will not be run. Contact Lotus Support to determine whether memcheck is necessary for your environment.

12.3.2 NSD explained

The NSD tool is constantly evolving. The following discussion applies to version 3.4.6.

NSD output is in plain text and can be viewed with any text file viewer.

The first section (shown in Example 12-1) contains a header with some basic information about the configuration of the machine.

Example 12-1 NSD output

```
Script Version : /opt/lotus/bin/nsd 3.4.6
Notes Version  : Release 5.0.8
Notes Base     : 5.08
Data Dir       : /data/notes1
Notes Exec Dir : /opt/lotus/notes/latest/sunspa
Search Path    : /opt/lotus/notes/latest/sunspa
                 /opt/lotus/notesapi
Debugger       : /usr/proc/bin/pstack
Debugger Version: Standard
MEMCHECK Version: MEMCHECK Version (4.4) for Lotus Notes Build V508 (July 21,
2001)
Script Dir     : /opt/lotus/bin
Host Info      : SunOS S588108PATC 5.8 Generic sun4u sparc SUNW,Ultra-4
User           : notes1 (notes1)
```

This is followed by the process tree. The process tree gives a listing of the Notes server processes and their parent/child relationship to each other.

In this example, the shell (sh) is listed as the parent for all processes, and the server is the parent of all Notes processes.

This can be useful, especially when there are orphaned Notes processes, as they will be represented with a return line between the other processes.

In the UNIX environment each process that is started has a parent process. In the case of a running Domino server, the parent of all the processes is the shell from which the server was started.

The first instance of the server process would be its child, and the server process would call other processes, such as event or update. These processes would be called the child processes of the server.

When one process exits, the child process for that process becomes “orphaned,” which means that the parent process has exited and the operating system reverts the parent to init, which is the first process started in a UNIX operating system. Init is responsible for loading all other processes and always has the process ID of 1.

This information can sometimes lead us to which process has crashed when a crash does occur and sufficient crash information is not captured (such as when the core file is truncated).

Example 12-2 is the process tree for a normally running server.

Example 12-2 NSD process tree

```
notes R    1071 sh
notes R    1097 /opt/lotus/notes/latest/sunspa/server
notes R    1114 /opt/lotus/notes/latest/sunspa/event
notes R    1099 /opt/lotus/notes/latest/sunspa/replica
notes R    1111 /opt/lotus/notes/latest/sunspa/adminp
notes R    1102 /opt/lotus/notes/latest/sunspa/replica
notes R    1112 /opt/lotus/notes/latest/sunspa/sched
notes R    1113 /opt/lotus/notes/latest/sunspa/calconn
notes R    1122 /opt/lotus/notes/latest/sunspa/clrepl
notes R    1104 /opt/lotus/notes/latest/sunspa/replica
notes R    1101 /opt/lotus/notes/latest/sunspa/replica
notes R    1098 /opt/lotus/notes/latest/sunspa/cladmin
notes R    1116 /opt/lotus/notes/latest/sunspa/cldbdir
```

```
notes R      1118 /opt/lotus/notes/latest/sunspa/clrepl
```

The R in the second column shows the process is active, running.

Example 12-3 is another process tree, this time for a server where the server process has exited without a trace.

Notice that none of these processes are shown as a parent/child to each other with one exception. Amgr is shown as being a parent for another amgr process. This is because the server loads amgr and the initial amgr process, then loads subsequent amgr tasks. All of the other processes were loaded as a child of the server process.

Example 12-3 NSD child process

```
nadmsup R    7551 /opt/lotus/notes/latest/sunspa/calconn
nadmsup R    7553 /opt/lotus/notes/latest/sunspa/http
nadmsup R    7549 /opt/lotus/notes/latest/sunspa/adminp
nadmsup R    7543 /opt/lotus/notes/latest/sunspa/update
nadmsup R    7545 /opt/lotus/notes/latest/sunspa/amgr
nadmsup R      7546 /opt/lotus/notes/latest/sunspa/amgr
nadmsup R    7540 /opt/lotus/notes/latest/sunspa/router
nadmsup R    7544 /opt/lotus/notes/latest/sunspa/stats
```

The next section contains the stack traces obtained from the debugger. These will probably not make much sense to anyone other than support and development.

The one thing you can check for is that one of the threads contains the word “fatal” or “panic.”

If the problem is a server crash and there is not a thread listed with either of those keywords, then it is likely that there was a problem during data collection and the crash information was not collected in time.

Note: There are exceptions to that rule, so always forward all available data to support, even if it appears to be of limited value.

For example, the following stack (Example 12-4) shows a fatal error on HTTP.

Example 12-4 NSD fatal error

```
#####
## thread 19/100 :: http pid=12634, k-id= 23495 , pthr-id=537117116
## stack      :: k-state=running, stk max-size=331772, cur-size=116812
#####
fatal_error(??) at 0xd10e7ac8
```



```
pthread_kill(??, ??) at 0xd0e7ad14
signal.raise(??) at 0xd0e7a94c
abort.abort() at 0xd0d79ca0
terminate.terminate__Fv() at 0xd0e8f2e0
invokedtr.__Invoke__Destructor(0x2000ea8c, 0x209501fc) at 0x1000cb58
```

Note: You should always see a `fatal_error()` call on the stack trace. This is the function that prints out the “Freezing all server threads...” message.

The next section will contain memcheck data if memcheck is installed.

The next section contains Inter Process Communication Facilities Status (IPCS) information. IPCS details the shared memory, message queues and semaphore information for the machine.

In R5, Domino on Solaris no longer uses System V shared memory, but instead uses `mmap()` files for shared memory. This change was made to increase performance.

The `mmap()` files will be listed here in the NSD under IPCS. Mmap files reside in `/tmp` and, similar to System V shared memory, have one control segment and several data segments. The data segments will be of uniform size, while the control segment is usually smaller than the data segments.

There is no need to manually remove these files on a successful shutdown of the server. If the server crashes, an `nsd -kill` will clean these files up.

You can also manually check for the existence of these files by issuing the command at the OS:

```
ls -la /tmp
```

Note: Each partitioned server will have its own set of mmap files. The owner of these files will be the user starting the different partitioned servers.

Here is an example:

Example 12-5 Viewing mmap files

```
@@@@@@@@@@@@@@@@@@@@ IPC STATS @@@@@@@@@@@@@@@@@@@@@@
-rw-r-- 1 nadmsup notes 3785776 Mar 20 10:16 /tmp/.NOTESMEM_please_do_not_remove.f802a800
-rw-r-- 1 nadmsup notes 0 Mar 20 10:15 /tmp/.NOTESMEM_please_do_not_remove.f802a800.LCK
-rw-r-- 1 nadmsup notes 8388608 Mar 20 10:16 /tmp/.NOTESMEM_please_do_not_remove.f802a801
-rw-r-- 1 nadmsup notes 8388608 Mar 20 10:15 /tmp/.NOTESMEM_please_do_not_remove.f802a802
-rw-r-- 1 nadmsup notes 8388608 Mar 20 10:16 /tmp/.NOTESMEM_please_do_not_remove.f802a803
IPC status from <running system> as of Mon Mar 20 10:17:07 EST 2000
T ID KEY MODE OWNER GROUP CREATOR CGROUP CBYTES QNUM QBYTES LSPID LRPID STIME RTIME CTIME
```

```

Message Queues:
T ID KEY MODE OWNER GROUP CREATOR CGROUP NATTCH SEGSZ CPID LPID ATIME DTIME CTIME
Shared Memory:
m 0 0x5007890d -rw-r-- root root root root 1 68 190 190 9:04:07 9:04:07 9:04:07
T ID KEY MODE OWNER GROUP CREATOR CGROUP NSEMS OTIME CTIME
Semaphores:
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

```

The next section is the Notes.ini followed by the Notes user's environment. If you have run the NSD as root then it will instead reflect the root user's environment.

Tip: Do not send the Notes.ini to Lotus Customer Support; it is included in the NSD output!

Next is the list of the Domino binaries directory. This list can help in determining if setuid root is in place for any processes, ownership of the binaries, and any add-ins that might be used on this server.

System information contains the hard and soft limits for the Notes user and the machine.

Hard limits are absolute limits set server-wide, which users cannot override, while *soft limits* pertain only to the users in whose environment the soft limits are set.

For instance, if the hard core limit is set to 2MB and the soft limit is set to 10MB for the Domino user, the Domino user will not be able to generate a core beyond 2MB in size.

Limits can be set in the file /etc/system using the **set** command.

For example, to set the hard limit for the number of file descriptors per process, in the /etc/system file enter the following line:

```
set rlim_fd_max=32768
```

Next, reboot the operating system. This will allow for enough file descriptors to run the server.

Table 12-1 on page 319 shows the hard and soft resource limits.

Table 12-1 Hard and soft resource limits

Parameter (and units)	Soft/current limits	Hard limits
Time (seconds)	unlimited	unlimited
File (blocks)	unlimited	unlimited
Data (kbytes)	unlimited	unlimited
Stack (kbytes)	8192.00	unlimited
Coredump (blocks)	unlimited	unlimited
Nofiles (descriptors)	64.00	1024.00
Memory (kbytes)	unlimited	unlimited

Swap info details the configuration of swap, as shown in Example 12-6.

Example 12-6 Swap information

Swap Info:				
=====				
swapfile	dev	swaplo	blocks	free
/dev/dsk/c0t0d0s1	32,121	16	2050432	2030368
total: 58216k bytes allocated + 39728k reserved = 97944k used, 1308752k available				

System Configuration shows devices configured for the server, followed by physical memory on the machine, number and types of processors, and some kernel configuration information.

Local Disks shows disk volumes mounted and space remaining, followed by current patches applied to the server. It is similar to a **df -k** command.

Example 12-7 Local disk information

Local Disks:					
=====					
Filesystem	kbytes	used	avail	capacity	Mounted on
/dev/dsk/c0t0d0s0	181969	38748	125025	24%	/
/dev/dsk/c0t0d0s4	963869	709048	196989	79%	/usr
/dev/dsk/c0t0d0s3	192833	23377	150173	14%	/var
/dev/dsk/c0t0d0s5	2025076	517179	1447145	27%	/opt
/dev/dsk/c2t0d0s3	8261430	278922	7899894	4%	/export/notesdata1
/dev/dsk/c2t1d0s3	8261430	236399	7942417	3%	/export/notesdata2
/dev/dsk/c3t0d0s3	8261430	237149	7941667	3%	/export/notesdata3
/dev/dsk/c2t2d0s3	8261430	9	8178807	1%	/export/logdata
/dev/dsk/c3t1d0s3	4131384	579316	3510755	15%	/export/home

Patches shows revision levels of patches currently applied to the machine.

Example 12-8 is an extract from the patches section.

Example 12-8 Patches information

```
Patches:
=====
Patch: 107430-01 Obsoletes:Requires: Incompatibles: Packages:SUNWwsr
Patch: 107063-01 Obsoletes:Requires: Incompatibles: Packages:SUNWtleux
```

VMstats shows a ten-second snapshot of the CPU statistics. This often shows plenty of idle time, since an NSD is most commonly gathered for a crash condition where the server is completely down.

This can be useful for performance and hangs, where the NSD is taken during the performance problem.

Network info gives a lot of information on the state of the network, and also lists the current connections to the server by IP.

Process list gives a complete list of system-wide processes. This can show other processes running in addition to Domino, and which could potentially be conflicting with the Domino Application.

For instance, if Apache (a public domain Web server) is running, you will be able to see that here. Since Apache and the HTTP process both default to running on port 80, they could potentially conflict with each other and cause problems.

Data directory gives a full listing of the data directory and subdirectories, as well as their ownership and access rights.

Note: You can check the size of your full-text index databases here.

Xwindows and Printer status are not often used; they are there for informational purposes.

The last section contains any errors that may have been generated by NSD during its execution. Some of these errors are really informational and not necessarily indicative of a problem, even if they are listed as a warning.

Generated Info/Warnings/Errors are shown in Example 12-9.

Example 12-9 Viewing messages

```
(1) INFO: New files added/delete to/from directory
      '/opt/lotus/notes/latest/sunspa'
(2) INFO: Generating binary list file /tmp/nsd.nadmsup/nsd_3.4.6_cache.ins.lst
```

```
(3) INFO: Generating cache file /tmp/nsd.nadmsup/nsd_3.4.6_cache.ins  
(4) WARNING: can't find the Notes memory checker binary 'memcheck5.0.8'
```

In this example the first informational message indicates that some files were changed within the binary directory. This may or may not be a problem, but is worth noting in case a process was overwritten or added recently which may be coinciding with problems on the server.

The next two informational messages are pointing to the location of temporary files which were created by NSD. The Warning listed here is telling us that the memcheck utility was not found along the search path.

Memcheck may not be installed on all machines since it is currently not installed as part of the Domino application.

12.4 ANSD tool

The Analyze NSD tool (ANSD) is a UNIX script based on awk and used to extract some basic information from an NSD-created file. It was created at EMEA Lotus Customer Support in Paris to automate the repetitive tasks of analyzing an NSD file. The authors are Marc Riart and Eric Dolce.

Note: awk is a pattern scanning and processing language. The acronym is derived from the initials of the three authors, Alfred V. Aho, Peter J. Weinberger, and Brian W. Kernighan. All the major UNIX systems have an installed version of awk.

12.4.1 How to use the tool

The ANSD tool reads a file created with the NSD tool and extracts some basic information from it.

Use the ANSD tool in the following way:

```
ansd [-help] nsd_file
```

ANSD provides the following information:

- ▶ Hostname
- ▶ Operating system
- ▶ Domino version
- ▶ Patches

- ▶ Kernel parameters
- ▶ Swap occupation
- ▶ Number of shared memory segments
- ▶ Stack of crashing thread
- ▶ Open databases by the culprit thread (provided that memcheck output is included within the NSD output file)

Regarding the Patches and Kernel parameters information, it compares the information found in the NSD output with the values requested by Lotus, typically the information that you can find in the Release Notes Domino database.

Here is an example for the Patches:

```
105050 is missing
103566 is at 25 (Recommended minimum 23)
103686 is at 01 (Recommended minimum 02)
103934 is at 04 (Recommended minimum 05)
```

Here is an example for the Kernel parameters:

```
Kernel parameters:
strctlsz is at 1024 ( Recommended 4096 )
v.v_proc is at 16218 ( Recommended 4096 )
```

Note: Both examples are made from a Domino 4.6x NSD output. There are not many patches and kernel parameters requests for Domino R5 on Solaris 8; consider them as an example to show the ANSD functionality.

12.4.2 Stack of the crashing thread

Maybe the most useful feature of the ANSD tool is the information on the Domino database that possibly caused the crash.

The tool looks for the fatal/Panic thread in the NSD file output and prints out the stack of the crashing thread. When the memcheck information is used in the NSD file, it shows which databases the crashing thread was using.

Example 12-10 shows a situation where the router process crashes with the mail.box file open.

Example 12-10 Viewing a crashed process output

```
Stack of crashing thread:
##### thread 1/8 :: router, pid=20709, lwp=1 #####
[1] eef37790 poll (0, 0, 3e8)
[2] ef097a04 unix_usleep (0, ed9be740, ef654d8c, 86871233, 3be60, ffff) + 4c
```

```
[3] 000165fc AddInMain (3c26c, 3c270, 35f7c, 37480, 0, 2566d0) + a04
[3] ef784f78 fatal_error (ee2a0630, ee2a0618, eecabae8, 0, 0, eecabe00) + 9c
[4] 0002261c NotesMain (1, 3cbb8, 3cbb8, 0, 0, 3cbb8) + 40
[5] 00022764 notes_main (0, 0, 0, 1, effff5a4, 0) + 104
[6] 00015be0 _start (0, 0, 0, 0, 0, 0) + dc
```

Open databases by the culprit thread [router:20709:1] :
/home/notesadm/notesr4/names.nsf
/home/notesadm/notesr4/mail.box

Note: Normally the names.nsf is still present in the open database list; any others indicate possible corrupted databases, in our example the mail.box database.

Be careful with this—the diagnosis that the open database is the cause of the crash depends strictly on the functions listed in the stack trace. This information can be confirmed only by an analyst from Lotus Support.

The ANSD tool can be used as a basic diagnostic tool, and can help you to recover from a server crash before the information from Lotus Support arrives.

In our example, the Domino administrator that reads the ANSD output can rename the mail.box database and restart the server.

You can also search the Lotus Knowledge Base for the names of the last functions before the fatal/panic functions to see if the problem is caused by a known bug that has been fixed in a new Domino version, or if a workaround exists.

Obtaining the ANSD tool

You can obtain the latest version of the ANSD tool from Lotus Customer Support, or by contacting the authors directly at their respective e-mail addresses: mriart@lotus.com and edolce@lotus.com.

12.5 The memcheck tool

Note: Memcheck is a utility developed by Iris which gathers information on the current state of the Domino memory pools.

12.5.1 NSD and memcheck

When NSD is run it will attempt to look for this program. If NSD finds memcheck it will by default run it against the server. The memcheck output will be contained as a section within the NSD log itself.

If memcheck finds an inconsistency it will also create a memcheck.dump file, which will contain an image of the memory areas which it had trouble with.

These files are only of use to Lotus Development. If created during the execution of an NSD, they should be forwarded to Lotus Support along with the NSD and console log. Memcheck is Domino version- and platform-specific.

Tip: A common mistake at customer sites is to install the memcheck without the execution file set. Check this after copying the file in the Domino binary directory; if it is not set correctly, use the **chmod +x memcheck** command.

12.5.2 Run memcheck manually

You can run memcheck manually in most instances. For details about the memcheck options run **memcheck -h**. This will yield information about usage of the command and available options.

If a crash occurs and the nsd-run memcheck does not get the information you require, memcheck can be forced to dump out an output file. It can also be forced to trigger on particular shared memory keys, or trigger on a particular process or set of processes.

One of the ways to do it is to run:

```
# memcheck -k <shared memory key> -d all -o <out file>
```

This attaches the shared memory key and dumps all the segments associated with the key manually. So, it would look like this:

```
# memcheck -k 0xf8000000 -d all -o memcheck.dump
```

Another way to run it is to run against a particular process known to be a problem:

```
# memcheck -P <pid> -d all -o <out file>
```

This will dump out the process's mem stack, as well as shared memory information (because the **-P** is a capital letter).

The **-o** out file option is available only with the **-d** option. Use the redirection to file character, **>**, to save the memcheck output for the other options. That should be sufficient for most instances.

The command **memcheck -h** can be run for more information on memcheck usage.

12.5.3 Open databases

One of the best features of the memcheck output is to show, for each Domino process/thread, which Domino databases are being used.

You have to search for the string “Open Databases” in the NSD output file; you can see some lines in Example 12-11.

Example 12-11 Finding open databases

```
/export/notesdata2/busytime.nsf
Version = 41.0
SizeLimit = 0, WarningThreshold = 0
ReplicaID = 852568A3:00745DD9
bContQueue = NSFPool[1:16804]
FDGHandle = 0xf018c0cf, RefCnt = 1, Dirty = N
DB Sem = (FRWSEM:0x0244) state=0, waiters=0, refcnt=0Writer = [0:0]
SemContQueue (FRWSEM:0x029d) state=0, waiters=0, refcnt=0Writer = [0:0]
By: [ sched: 6016: 2] DBH= 33, User=CN=server2/0=Saturn
```

The example shows that the **busytime.nsf** database is opened by the sched process **6016** thread **2**.

This information can be matched with an eventual stack trace of a faulting thread to produce a preliminary diagnosis on an eventual corrupted database that generated the server crash.

Note: This is a possible starting point to troubleshoot the problem. A confirmation from Lotus Customer Support must also be done, however.

This is also how the ANSD tool works to find the open databases by the crashing thread.

12.5.4 Memcheck.dump file

What does it mean exactly when the memcheck tool shows warnings like the one below and it dumps a memcheck.dump file?

Example:Warnings from memcheck

```
WARNING (57): Skip POOL (0x0414) fragment size=4, @748140
WARNING (57): Skip POOL (0x0414) fragment size=4, @748144
WARNING (33): Invalid POOL (0x0414) free chain (next=45055) @748148
WARNING (32): Invalid POOL (0x0414) free end signature 0x0000 @813676
** Dumping shared DP00L '0xf8ab1001' @0xed000000 to
./memcheck.SUNSPA_iena_03_17
```

@20_25.dump

Normally this should never happen. This could be a problem with the memcheck tool itself. If you experience a memcheck.dump you should contact Lotus Support.

12.5.5 Memory dump commands

From the Domino server console, type the following command:

> show memory dump

This will display the available memory (including virtual) on the server console and also create a memory.dmp file in the Domino data directory.

Tip: The same things can be done using the **server -m** command from a UNIX prompt. Pay attention, in this case you have to **kill -9** the process manually.

Add this file to the list of files to send to Lotus Customer Support.

12.6 Core file

There is a great deal of confusion in the user community as to when core files should be generated, what Killprocess is, and when it should be removed from the Notes.ini.

Core files are OS-level dumps of a faulted process. By default Domino will not generate a core file, but it can be configured to generate one in a crash. The core image is written in the process's working directory, and it contains all the process information pertinent to debugging: contents of hardware registers, process status, and process data.

The core file is generally of use only to the code developers, but sometimes the end user can find helpful information in it to troubleshoot the problem.

Many administrators enable core file generation and leave it enabled on the assumption that more information is better; however, typically it is best to leave core file generation disabled on a server which is not currently experiencing any problems.

Core files can be rather large and unwieldy. They tend to consume a lot of hard drive space, and transmitting them to Lotus Support can be time-consuming. OS-imposed limits on an improperly configured server can also truncate a core, leaving the trace from the core useless. In those cases we will have less information to diagnose the problem than we would if core generation were disabled.

For a “typical” crash the information that NSD gathers is sufficient for diagnosis. Core file generation may be requested by Lotus if more in-depth information is needed from a recurring crash or hang.

To enable core file generation for Domino, put the following environment variable in the Notes user’s environment:

DEBUG_ENABLE_CORE=1

Also make sure your ulimits are set correctly: they should allow for at least 100 MB file size.

Some administrators set both to “unlimited,” but this can be dangerous since it allows the core file to grow to the size of the available disk space.

To check the current limit use the **ulimit** command, shown in Example 12-12.

Example 12-12 Using the ulimit command

```
$ ulimit -a
time(seconds) unlimited
file(blocks) unlimited
data(kbytes) unlimited
stack(kbytes) 8192
coredump(blocks) unlimited
nofiles(descriptors) 64
memory(kbytes) unlimited
```

To check the soft/hard limits use the **sysdef** command, shown in Example 12-13.

Example 12-13 Using the sysdef command

```
$ sysdef
(Example output...):
Infinity:Infinity    cpu time
Infinity:Infinity    file size
7ffff000:7ffff000    heap size
```

```
800000:7ffff000    stack size
Infinity:Infinity   core file size
40: 400    file descriptors
Infinity:Infinity   mapped memory
```

To set the soft limit of the core file as the Notes user:

```
$ ulimit -S -c <SIZE>
```

This is valid only for the Notes session.

To set the hard limit of the core file as the Notes user:

```
$ ulimit -H -c <SIZE>
```

The *soft limit* is the current limit set for a resource, and the hard limit is the maximum limit that can be set for the resource on the Solaris OS. For more information on the `limit` and `ulimit` commands consult the Solaris product documentation.

12.6.1 NSD and the core file

If a core file exists in the Domino data directory, the NSD tool will rename the core file, putting in the following information that it extracted from the core file:

- ▶ Domino process name
- ▶ UNIX platform
- ▶ Hostname
- ▶ Date and Time at core generation time

For example:

```
core.icm.SUNSPA_iena_03_09@00_37
```

The NSD automatically compresses the core file, using the standard UNIX compress utility, so you will find the core file name with a `.Z` extension:

```
core.icm.SUNSPA_iena_03_09@00_37.Z
```

12.6.2 The gcore command

After enabling core file generation on the UNIX Solaris operating system by setting `DEBUG_ENABLE_CORE` in the server account's user environment, the Domino server process still does not generate a core file.

You can force/generate a core file of any Domino process by identifying the process ID that produced a fatal error, or that is currently hung. You should then issue the following UNIX command:

```
/usr/bin/gcore <process.id>
```

You must be the root user to use this command. The core file will be generated in the directory from which the command is issued and will be named “core.pid,” where pid is the process ID of the dead process. Before using this command, make sure that the server is either in a crash state, or a truly hung state.

Also, be careful that your file system can contain the core file; sometimes the size of the core file can be hundreds of MB.

Note: **gcore** is a command currently only found under Solaris. Neither AIX nor HP has an equivalent.

12.6.3 How to read a core file

The format of the core file is not ASCII, so you will need to use one of several tools available to extract useful information from it.

The **adb** tool

The **adb** utility is an interactive, general-purpose debugger. It can be used to examine files and provides a controlled environment for the execution of programs. You can find it in all of your UNIX systems.

In this example we use it to examine a core file generated by a Domino process. Normally the core file is generated in the Domino data directory.

The name of the core file is simply “core.” To find which process made the core file, use the file command, for example:

```
# file core  
core: ELF 32-bit MSB core file SPARC Version 1, from 'icm'
```

Once you know which process generated the core file, use the **adb** tool to list the stack trace of the faulting thread.

To do this you have to start **adb** with 2 parameters, the process name and the core file. For the process you have to give the absolute pathname. All the Domino processes are in the directories /opt/lotus/notes/latest/sunspa, so if the process that generated the core file is **icm**, the command is:

```
# adb /opt/lotus/notes/latest/sunspa/icm core  
core file = core - program ``icm'' on platform SUNW,Ultra-4  
SIGABRT: Abort
```

The adb output shows the signal received by the process that caused the core dump file (Abort in our example).

Note: The adb is an interactive tool. By default it does not return a prompt. If you want a prompt, use the **-P** prompt option.

At this point you can list the stack trace with the **\$c** command, for example:

```
adb> $c
__lwp_kill() + 8
__sighndlr(a,fd2f1b50,fd2f1898,fd1755ac,fd2f1e4c,fd2f1e2c) + c
sigacthandler(a,fd2f1b50,fd2f1898,fe13c524,28,fd2f1e5c) + 748
__0fIICMStatsQGatherStatisticsUsT(3000,403a0,1,28,fd2f1c30,8000) + 164
__0fHCICMSrvJStatsTaski(3f4e8,1000,3000,1388,ea60,0) + fc
ThreadWrapper(0,252c8,0,fe13db40,1,fe13c524) + 8c
```

Note: We used the **-P "adb>"** to have the adb> prompt in the command.

In this example the possible function that created the crash was `0fIICMStatsQGatherStatisticsUsT()`. Use this string to look in the Lotus Knowledge Base to gather more information about the crash.

Exit from adb with the command **\$q**. See the online manual for more information on adb.

The strings command

The **strings** command finds printable strings in an object or binary file. It is useful for identifying random object files and many other things.

In this example we can use it to identify which process made the core file by using the following command:

```
# strings core | more
CORE
http
/opt/lotus/notes/latest/sunspa/http
```

You should find the name of the faulting process in the first lines (in our example it is the HTTP process).

You can redirect the strings output into a file so you can examine it using your preferred editor:

```
# strings core > core.txt
```

The memcheck tool

You can also use the Domino memcheck tool to read a core file. In this case the memcheck tool extracts the information about the memory mapped files used by Domino to share information between the Domino processes (the files called .NOTESMEM_please_do_not_remove.f8xxxx in the /tmp directory).

To do this use the **-c** option of the memcheck tool, along with the maximum verbose level of memcheck, **-v5**, and redirect the output into a text file:

```
# memcheck -v5 -c core > core.txt
```

You can find some useful information in the created text file, such as:

- ▶ The process heap memory
- ▶ The handle table descriptor
- ▶ The names of some Domino databases that are possibly causing the problem

Again, the best people to analyze this file are Lotus customer support analysts and IRIS developers.

12.6.4 Killprocess

Killprocess is a Notes.ini parameter, which when set to 1 will attempt to take a Notes partition down entirely when it detects a crash condition.

The intent of this parameter is to simplify the cleanup process on a partitioned server where multiple partitions share the same UNIX account. It has been found that the best way to handle partitioned servers is to create a separate UNIX account for each partition.

This makes taking down individual partitions and diagnosing problems much easier. When separate accounts are created for each partition this parameter should be manually removed from the Notes.ini.

It is also worth noting that this parameter is added to the Notes.ini when “Domino Enterprise Server” is selected on install, and unless needed it should be removed.

12.7 Troubleshooting

12.7.1 Crashes

When a server crashes, the following output is typically displayed on the server console:

```
Fatal Error signal = 0x0000000b PID/TID = 8107/1  
Freezing all server threads ..
```

This message shows that the server had a hard crash and was forced to exit.

Freezing all server threads means that Domino has put all of its threads into a spin and will not continue until the application environment is cleaned up and restarted.

This message also shows us the hexadecimal equivalent of the signal that it exited on (0x0000000b or Segmentation Violation) and also shows the process ID and the thread ID of the thread which encountered the exception. In this case it was process number 8107, thread number one.

Running an NSD with no options against this server will gather the crash level information. Following is the thread for this crash. Note that the word “fatal_error” is contained within the stack. This will be contained in any threads which have encountered an application exception.

Threads are always read from the top down. The top of the stack represents the last thing the server was doing and the bottom is the beginning of the thread.

In our example the cause is not so obvious. It shows a normally running thread which goes into a `unix_usleep`, which is simply the thread waiting for something to happen.

It then goes into a `sigacthandler`, which is the signal handler that is called when a signal is generated indicating a problem.

In this case the real problem is that the `amgr` task was killed off by the administrator by issuing a `kill -11 8107` at the operating system command line.

This forced the crash condition shown. Typically the crash will occur due to an error within a function that is being executed by the thread. This example is a good way to show how a process external to Notes can force the server to crash.

To find out which signal the thread crashed on, convert the hexadecimal number to a decimal number and look it up in the file `/usr/include/sys/errno.h`.

In this example the hexadecimal number was 0b, which is 11 in decimal. The signal.h file shows the following:

Example 12-14 Signal.h header file

```
#define SIGHUP 1 /* hangup */
#define SIGINT 2 /* interrupt (rubout) */
#define SIGQUIT 3 /* quit (ASCII FS) */
#define SIGILL 4 /* illegal instruction (not reset when caught) */
#define SIGTRAP 5 /* trace trap (not reset when caught) */
#define SIGIOT 6 /* IOT instruction */
#define SIGABRT 6 /* used by abort, replace SIGIOT in the future */
#define SIGEMT 7 /* EMT instruction */
#define SIGFPE 8 /* floating point exception */
#define SIGKILL 9 /* kill (cannot be caught or ignored) */
#define SIGBUS 10 /* bus error */
#define SIGSEGV 11 /* segmentation violation */
```

Here is the thread stack trace:

```
#####
##### thread 1/6 :: amgr, pid=8107, lwp=1, tid=1 #####
#####
[1] ff2962b0 lwp_sema_wait (30a40)
[2] fe3cb030 _park (30990, 30a40, 0, 1, fe3ed228, 0) + 10c
[3] fe3cad24 _swtch (5, fe3ec4b4, 30a20, 30a1c, 30a18, 30a14) + 338
[4] fe3c964c cond_timedwait (ffbeede0, ffbeedc8, ffbeeda0, 30990, fe3ec4b4, 0) + 1f4
[5] fe3d97c0 sleep (1e, 0, fe3ec4b4, 1e, 5f5e100, 0) + 28c
[6] fe4bd2c0 fatal_error (b, fedd8ebc, 1fab, fe3ec4b4, 30a14, 309f4) + 184
[7] fe3d8eec __libthread_segvdhldr (b, ffbef340, ffbef088, ffbeefc8, fe3ec4b4, 0) + e0
[8] fe3db8b0 __sigdhndlr (fe3d8e0c, b, ffbef340, ffbef088, ffbeefc8, 309f4) + 10
[9] fe3d82cc sigacthandler (b, ffbef340, 30990, fe3ec4b4, ffbef088, fe3d8e0c) + 71c
[10] fe4e4fd4 unix_usleep (f4240, 1, ffffffff, ffffffff, 37f9c, 8000) + 7c
[11] fe4b4188 AddInIdleDelay (3e8, 37f70, 37f70, 330e, 0, 64) + 4c
[12] 0001a428 ManagerMain (302d8, 3000, 30454, 0, 0, 20) + 138
[13] 0001a2e0 AddInMain (30494, 1, 323a4, 0, 0, 0) + 1f8
[14] 0001f090 NotesMain (1, 323a4, 323a4, 0, 0, 6c) + 40
[15] 0001f1d8 notes_main (0, 0, 0, 1, ffbebf14, 0) + 104
[16] 000152a0 _start (0, 0, 0, 0, 0, 0) + dc
```

Tip: Using the Solaris **pstack** command is a simpler way than NSD offers to see what the call stacks are at a given moment. The **pstack** command prints a hex+symbolic stack trace for each lwp in each process. Use the command:
/usr/proc/bin/pstack PID

12.7.2 Fault recovery

Fault recovery is a method for automatically restarting a Domino server when it crashes. With fault recovery it is possible to automatically gather diagnostic data and recycle a Notes server when it encounters a fatal condition.

Fault recovery relies on a UNIX signal to be generated and therefore will not allow you to recover from a hang condition. If there are any errors within the fault recovery script, fault recovery will not take place. Therefore, it is important to keep your cleanup script simple initially.

To enable fault recovery set the Notes.ini variable `cleanupscriptpath=` to point to the directory containing the script you want Domino to execute on a fatal event.

The Notes.ini parameter `Fault Recovery=1` must also be set.

Both Fault Recovery and Killprocess rely on the same set of UNIX signals which are generated to indicate a problem. The table below lists those signals. Since both killprocess and fault recovery intercept the same signals, they cannot be used together.

Table 12-2 Fault recovery and kill process signals

Signal#	Name	Description
0x00000004	SIGILL - ill	Illegal Instruction
0x00000006	SIGIOT-iot	IOT instruction
0x00000007	SIGEMT- emt	EMT instruction
0x00000008	SIGFPE- fpe	Floating Point Exception
0x0000000A	SIGBUS- bus	Bus Error
0x0000000B	SIGSEGV- segv	Segmentation Violation

With fault recovery set, the server will automatically clean up after a crash, but will not allow you to run any diagnostics.

Example 12-15 on page 335 is a fault recovery script which *will* allow for automatic recovery of crashes and at the same time gather the crash information that can be forwarded on to Lotus Support for analysis.

The end result of this script is to create an NSD with crash information, create a new directory beneath the /log directory with a time/date stamp in the title, and move the NSD log as well as the NSD -kill log and the console log to the newly created directory.

It goes on to create a tape archive of these files and places them in the /logs directory with an appropriate title. In this case if the crash occurred at 4:30 am on December 5th the archive would be called crash_file_12_05@4_30.tar. The script then exits and Domino restarts.

The directory structure with the date stamp cleans up the Notes data directory significantly and makes it easier for the customer to provide the required information to Support.

Example 12-15 Fault recovery script

```
#!/usr/bin/ksh
NOW=`date +%m_`$d@%H_%M`
#
#call nsd to get preliminary crash informaton
#
/opt/lotus/bin/nsd
#
# Make a directory unique to this outage for data collection
#
mkdir /logs/$NOW
#
#Move the data collected to the directory we just created
mv /notes/data/console.log /logs/$NOW/
mv /notes/data/nsd_* /logs/$NOW/
#
#tar up the files we collected to make it easier to send the data to support
#
tar cvf /logs/'crash_file_'$NOW'.tar' /logs/$NOW/
exit
```

12.7.3 Planning your startup and shutdown scripts

Startup scripts are not provided by Lotus because each site will have very specific needs and fundamental differences in their environment.

While it is possible to launch the server without using one, a startup script can go a long way in preventing problems. The console output is by far the most important reason to employ one. Many status and error messages are sent to the console which are not recorded in the Notes log. This information can be invaluable in diagnosing a problem.

The first rule is to keep the script simple until you are sure it works properly. Basic functionality is far more important than adding all the bells and whistles you can think of.

In its most basic form a script for starting a server would look like the one shown in Example 12-16.

Example 12-16 Basic Domino server startup script

```
#!/usr/bin/ksh
#
#
# Script to start the Notes server
#
# start the server and redirecting all input and output to files.
#
/opt/lotus/bin/server < /notesuser/notesr4/console.in >
/notesuser/notesr4/console.out &
#
```

This script will start the server, and specifies an input file from which it will read the console commands (console.in) and a log file which will contain all of the console output (console.out).

Commands can be echoed to the input file and would be accepted as if they were entered directly from the console.

Example:

```
# echo 'sh tasks' >> console.in
```

Note: For more information on startup/shutdown scripts, see 3.5, “Starting the Domino server” on page 81 and 3.6, “Shutting down the Domino server” on page 85.

An effective shutdown script (such as the one in Example 12-17) is also important. Many administrators shut down their servers via scripts to do unattended backups of their servers. It is important to construct shutdown scripts carefully because a faulty shutdown script, or one that fails to anticipate a server hang on shutdown, may incur more downtime.

Example 12-17 Domino shutdown script

```
#!/bin/ksh
#
# Shutdown script for a Solaris server
#
# Issue a server -q command to try and stop the server gracefully
# Note that you must background this command or this script will hang
# if the server does not exit gracefully
#
/opt/lotus/bin/server -q &
#
```

```

# Sleep for 5 minutes to allow Notes time to shutdown
# This can be changed to reflect a shorter time period but you should give
# Notes at least 2 minutes to finish shutting down
#
sleep 300
#
# Now check the process table to see if the Notes server process is still running
# This part is platform dependent so be sure to change the /sunspa/ directory
# to the appropriate platform directory for the server (hppa,ibmpow, or sunspa)
#
SERVERUP=`ps -ef | grep /opt/lotus/notes/latest/sunspa/server | grep -v grep | awk
'{print$8}'`
#
# Test to see if the server is still up
#
if [ $SERVERUP ]
then
    # Server is still up so we will issue a nsd -kill
    #
    echo 'server is still up after 5 minutes...'
    echo
    echo 'taking a NSD'
    /opt/lotus/bin/nsd
    echo 'shutting down server with nsd -kill'
    /opt/lotus/bin/nsd -kill
else
    #Server has shut down normally no nsd -kill needed
    #
    echo 'server has shut down gracefully'
fi
exit

```

This script could conceivably do a lot more to help in the data collection process. We could combine some of the things we did in our fault recovery script to create a directory for the data gathered, and perhaps notify the administrator via e-mail that there was a problem.

The most important functions were accomplished: the server was back up within a short period of time, and the data that was needed to figure out why the server hung was preserved.

Note: For a more detailed startup/shutdown script see Appendix F, “Example script to start and shut down a Domino server” on page 395

12.8 Using the debug_XXX variables

Domino server uses a lot of debug variables to augment the trace level of its activities.

In this section we discuss the debug variables used most commonly to troubleshoot a crash or hang problem.

12.8.1 How to use

The first step in implementing this parameter is to add an entry to the Notes.ini file, as in the following example:

```
DEBUG_OUTFILE=/tmp/DebugDomino.log
```

The variable `DEBUG_OUTFILE` contains the filename where the debug information will be stored.

Tip: To view the output of the debug outfile on the screen, use the UNIX command `tail -f /tmp/DebugDomino.log`. The screen will show the last lines of the file.

A Domino debug variable is normally recognized by the prefix `DEBUG_`, for example `DEBUG_THREADID=1`.

Remember that the content of the Notes.ini file is not case-sensitive.

12.8.2 Performance issues

Set the following two parameters in the Notes.ini file:

```
DEBUG_CAPTURE_TIMEOUT=1  
DEBUG_SHOW_TIMEOUT=1
```

These two parameters are necessary for capturing additional performance details, enabling Lotus Support to determine if a slowdown is due to some threads holding on to a semaphore for a long period of time. There should be little impact on performance when setting these parameters.

When a server slowdown occurs, any semaphore time-out data which is traced via these two debug parameters is written into a file named `SEMDEBUG.TXT`, located in the Domino data directory, and the Domino console log file (text logging, *not* log.nsf).

Note: The SEMDEBUG.TXT file is created only if a semaphore time-out occurs. Any time this happens, collect an NSD output and send this file to Lotus Customer Support.

Output of SEMDEBUG.TXT for UNIX

```
THREAD [01676:00001] WAITING FOR RWSEM 0x412C (@EE100210)
(R=0,W=1,WRITER=05067:00001,1STREADER=05067:00001) FOR 30000 ms
THREAD [01684:00001] WAITING FOR RWSEM 0x412C (@EE100210)
(R=0,W=1,WRITER=05067:00001,1STREADER=05067:00001) FOR 30000 ms
```

What does the output mean?

The output shown has the following meaning:

- ▶ 0x412C indicates the type of semaphore.
- ▶ [01684:00001] The first number (01684) indicates the process ID. The second number (00001) is the thread ID.

You can match this number with the NSD stack trace. This is shown in Example 12-18.

Example 12-18 Finding the cause of a system slowdown NSD output

```
[1] 1684:/opt/lotus/notes/latest/sunspa/tmmScan <-- fatal thread
##### thread 1/4 :: tmmScan, pid=1684, lwp=1 #####
[1] eed396a0 lwp_sema_p (228a30)
[2] eed396a0 __lwp_sema_wait (228a30, 1d670, 0, 0, 0, 0) + 8
[3] eefc779c _park (228990, 228a30, 0, 1, eeef6240, 0) + a0
[4] eefc7554 _swtch (2289a0, 228b90, 228a10, 228a0c, 228a08, 228a04) + 2cc
[5] eefc5f8c _cond_timedwait_cancel (efffdd90, efffdd78, efffdd70, 228990, eeef52b0, 0) + 1e4
[6] eefd3420 _ti_sleep (1e, eeef52b0, eeef52b0, effff2e3, effff154, f8000600) + 100
[7] ef0776e4 fatal_error (b, ef663dd4, ef657598, efffde50, 228a04, 2289e4) + 2c0
[8] eefd2f20 __libthread_segvdhr (b, efffe4b0, efffe1f8, efffe138, eeef52b0, 2289e4) + e0
[9] eefd2334 sigacthandler (b, efffe4b0, 228990, eeef52b0, efffe1f8, eefd2e40) + 6e0
[10] 000201d0 KC_ScanStart (efffeec0, 3ae18, efffeec0, efffeec4, efffeec8, efffeec8) + 13a0
[11] 000157b0 AddInMain (22, 1, effff144, ef7ed2b8, ef7ec9c0, 0) + d00
[12] 0002fb40 NotesMain (1, effff144, effff144, 0, 0, ef7c16e1) + 40
[13] 0002fa60 notes_main (0, 0, 0, 1, effff144, 0) + a8
[14] 000147ec _start (0, 0, 0, 0, 0, 0) + dc
```

In this way you can identify the thread that caused the semaphore timeout issue.

You can also check a semaphore timeout at the server console reading the Domino statistics. To view this statistic, type the following command at the server console:

```
sh stat sem.timeouts
```

If a semaphore timeout occurred, you will see a Sem.Timeouts statistic like the following:

```
Sem.Timeouts = 430D:58 0A13:42 030B:28 0116:26 0A12:21
```

Note: A single semaphore timeout is not always a symptom of performance issues. You should only be concerned if you experience a lot of them.

12.8.3 Typical debug scenario

A Domino administrator has experienced multiple server crashes and has seen the following panic messages for the last three crashes:

From crash #1:

```
Thread=[01684:00001]  
PANIC: LookupHandle: handle not allocated
```

From crash #2:

```
Thread=[01688:00001]  
PANIC: LookupHandle: handle not allocated
```

From crash #3:

```
Thread=[01691:00001]  
PANIC: LookupHandle: handle not allocated
```

You add the Notes.ini parameters mentioned previously, and when the server crashes again you should be able to locate the following information from the debug file that was specified by `DEBUG_OUTFILE=/tmp/DominoDebug.txt`.

From the DominoDebug.txt file from crash #4:

```
[01695:00001] Thread=[01695:00001]  
PANIC: LookupHandle: handle not allocated
```

Searching through the Debug (DominoDebug.txt) File, you find:

```
[01695:00001] 03/06/98 08:07:09 AM AMgr: Start executing agent 'Update  
DBCatalog' in 'admin\notes\DbCatalog.nsf' by Executive '1'
```

When the server crashes again, you should be able to locate the following information from the debug file that was specified:

From the DominoDebug.txtfile from crash #5:

```
[01696:00001] Thread=[01696:00001]  
PANIC: LookupHandle: handle not allocated
```

Searching through the debug (DominoDebug.txt) file, you find:


```
[01696:00001] 03/06/98 10:49:17 AM AMgr: Start executing agent 'Update
DBCatalog' in 'admin\notes\DbCatalog.nsf' by Executive '1'
```

Because of the `DEBUG_THREADID=1` parameter in the `Notes.ini`, you should be able to pinpoint the cause of the crash to an agent that is being run. Disabling the agent may resolve the crash.

12.9 How to prevent a crash

In this section we present some recommendations for avoiding crashes due to a corrupted database.

12.9.1 Maintenance policies

Running **fixup**, **upda11**, and **compact** on a scheduled basis can prevent problems from occurring on the server due to a program document. However, there is a fine line between using these utilities to prevent problems and actually creating problems by using them when it is inappropriate.

Fixup

Fixup will attempt to detect and fix database corruption.

At first it may seem like a good idea to run **fixup** nightly to try and prevent problems from occurring. However, **fixup** is disk-intensive, and careful planning should be done to ensure that any system-wide running of **fixup** is scheduled when user activity is at a low point, and that no other disk-intensive processes or applications (such as a backup utility) are running at the same time.

If the pool of databases that **fixup** is to run on is too large, **fixup** could end up running well into the next period of busy activity for the server.

Upda11

Upda11 will attempt to update all of the views in a database and full-text indexes. When run with the option of **-r** it will discard all of the view indexes and rebuild them from scratch.

While this does not result in a loss of information, it is also extremely disk-intensive, and careful planning is necessary to avoid contention with other processes.

Compact

Compact will attempt to improve the storage efficiency of the database on which it is run. When documents are deleted from a database the space which they used to occupy is left as “white space,” which is reused by the server when documents are added or expanded.

At times this can mean that a database is left with a considerable amount of unused white space, which can be reclaimed for other databases.

In R4 this was accomplished by making a copy of the database in a temporary file, deleting the old database, and moving the temporary copy in as a replacement for the original database. This also meant that users were not allowed to access the compacted database until the operation was completed.

In R5 Domino is now able to do a compaction on a database without making the copy. This is much more efficient, and also allows you to reclaim white space on a file system which has run out of space entirely.

Compact will not fix database corruption, however, and should be run only when the goal is to improve the efficiency of the server, not to attempt to prevent a crash due to corruption. Many sites have chosen to enable this on a weekly basis with an **-s** flag to tell **compact** to run only against databases with a certain threshold amount of white space.

12.9.2 Using NSD

Run the NSD script regularly to monitor your Domino system.

We recommend that you use the **-info** option, to avoid attaching to the process threads in a Domino production environment. Attaching to the process could lead to a crash or hang on a normally running server.

In the resulting file check the following:

- ▶ Size and number of memory mapped files
- ▶ Operating system details
- ▶ Network statistics
- ▶ Date and size of the full-text index directories

Tip: You can do this regularly, for example once a week, using the **crontab** UNIX command.

12.9.3 Avoid full-text indexes for names and log databases

Some Domino administrators make a full-text index version of their Domino Directory database names.nsf.

This is not really recommended, since it can result in a server crash or poor performance.

The reason is that the names.nsf database is accessed heavily by all the Domino processes. The files created by the FT Index, under the directory named names.ft in this case, can become corrupted and this can lead to a server crash.

Note: This is much more true on Domino V4, but it can happen in Domino V5, too.

If you want to use the FT Index feature you should make an off-line copy of your names.nsf and use this as a full-text index database.

The same precautions against using the FT Index feature are valid for the log.nsf database.

12.9.4 Collecting statistics

This section defines some basic fields in a statistics report to help you analyze the performance of a Domino server.

Stats.Time.Current

This is the time that the statistic report was recorded.

Stats.Time.Start

This is an important statistic. In order to gauge other statistics on the server, it is important to know how long the server has been running. (For instance, 100 semaphore timeouts for a server that has been running for two weeks is not as alarming as if the server has been running for one day.)

Tip: Use the field Reporter.Time.Elapsed to see how long the server has been running. This is easier than using Stats.Time.Current and Stats.Time.Start and subtracting the two figures.

Database.DbCache.OvercrowdingRejections

This represents the number of times a database was not put into the cache when closed because `DbCache.CurrentEntries => Database.DbCache.Maxentries`. If this number gets too large, it is possible to add a `Notes.ini` parameter to increase this setting. Overcrowding rejections are common on a busy server and can get very high. If a server is experiencing less than 50 a day, this is probably not a problem. However, several thousand in a day is something that should be investigated.

Database.DbCache.MaxEntries

This is the number of databases that the server can hold in its cache at once. This number defaults to 25 or `NSF_Buffer_Pool/300K`, whichever is greater. The number of databases held in cache is 1.5 times what the `Maxentries` is set to. In order to adjust the `Maxentries`, add the following to the `Notes.ini`:

`NSF_DbCache_Maxentries=`

Note: In Notes 4.x, the maximum value for `NSF_DbCache_MaxEntries` is approximately 723, depending on the platform.

Sem.Timeouts

Do any `Sem.Timeouts` exist? How long has the server been up? It is not uncommon for semaphore timeouts to occur on a busy server. More than 10 timeouts of a particular type (i.e. 0244:20) should be noted; however, it may not be a problem. Several hundred in a day is something that should be investigated. The statistic `Sem.timeouts` will not appear in the `statrep` if the server has not experienced any semaphore timeouts.

Server.Task

What is running? Are there any hex sessions? A few hex sessions are not uncommon on a busy server. What to look for here is a large number of hex sessions. Also, if by looking at the other statistics on the server, you determine that this is not a busy server and there is a large number of hex sessions, this may indicate a corrupt `names.nsf`. What third-party applications are listed in this field? Is the server running multiple instances of one task (for example, multiple updaters, replicators, and so forth)?

Server.Users

This field shows how many users are on the server at the present time.

Server.Users.Peak

This statistic indicates the peak number of concurrent users since the report task was started. This is a good statistic to quickly gauge if the server is being pushed beyond its capabilities with the current configuration. For example, a server with 256 MB RAM with 500 concurrent users most likely needs more RAM and is in danger of becoming an unstable server.

Tip: The Notes.ini parameter Platform_Statistics_Enabled was introduced in Domino Release 5.0.2. This parameter reports some OS level and Domino process statistics using the command **show stat platform**. At the time of this writing, this feature is valid only on Solaris and Windows NT platforms.

12.9.5 Show transaction command

The **Show Transactions** command is mainly used for troubleshooting purposes. It produces the following information:

- ▶ Count - The number of these transactions that have executed since startup
- ▶ Min - The minimum msec that one of these transactions has taken to execute
- ▶ Max - The maximum msec that one of these transactions has taken to execute
- ▶ Total - The total msec that all of these transactions have taken
- ▶ Average - The average msec that all of these transactions have taken

12.9.6 File descriptor issue

The number of file descriptors used by the Domino processes should be monitored to prevent hanging problems for these processes if the system limits are approached.

To find the File descriptor usage for a particular process, it is a simple case of running the Solaris command:

```
#/usr/proc/bin/pfiles <PID>
```

where <PID> is the Process ID of the process about which you are interested in obtaining data from running the **ps** command. To find the total number of FD in use without counting, issue the command:

```
/usr/proc/bin/pfiles <PID> | grep mode | wc -l
```

Another very useful tool is the **lsof** (LiSt Open File) command. This is a free tool that you can download from the Sun website at www.sunfreeware.com/

You can use it with the `-p <PID>` option to select the process that you want to check:

```
# lsof -p <PID>
```

Note: The `lsof` tool is used by the NSD script, too, using the `-lsof` option.

12.10 Solaris tools

There are several useful commands in Solaris for troubleshooting application problems. They are:

- ▶ **truss** - Trace system calls and signals
- ▶ **sotruss** - Trace shared library procedure calls
- ▶ **snoop** - Capture and inspect network packets

12.10.1 The truss utility

The **truss** utility executes the specified command and produces a trace of the system calls it performs, the signals it receives, and the machine faults it incurs.

This command is useful for troubleshooting Domino process hangs. If you suspect that one process is hanging, use the following command, with the `-p pid` option, to trace which system calls the process is using.

```
# truss -o outputfile -p pid
```

Tip: You must use the `-o outputfile` option as the first option, otherwise it does not work.

To stop collecting the information, use **Ctrl-C**.

Example 12-19 shows `truss` output in the HTTP process.

Example 12-19 Output from truss for the HTTP process

```
setitimer(ITIMER_REAL, 0xFD909780, 0x00000000) = 0
setcontext(0xFD909A18)
sigprocmask(SIG_BLOCK, 0xFDF8DECO, 0x00000000) = 0
setitimer(ITIMER_REAL, 0xFD909CC0, 0x00000000) = 0
Lwp_sema_post(0xFDF8DEDO) = 0
sigprocmask(SIG_UNBLOCK, 0xFDF8DECO, 0x00000000) = 0
Lwp_sema_wait(0xFDF8DEDO) = 0
sigprocmask(SIG_BLOCK, 0xFDF8DECO, 0x00000000) = 0
setitimer(ITIMER_REAL, 0xFD909CC0, 0x00000000) = 0
sigprocmask(SIG_UNBLOCK, 0xFDF8DECO, 0x00000000) = 0
```

```
nanosleep(0xF2705C90, 0xF2705C98)      = 0
Time()                                   = 953332451
Time()                                   = 953332451
nanosleep(0xF2807BE8, 0xF2807BF0)      = 0
Time()                                   = 953332451
poll(0xF2501B08, 1, 5000)               = 0
nanosleep(0xF2705C90, 0xF2705C98)      = 0
Time()                                   = 953332452
Time()                                   = 953332452
```

Reading the truss output is not simple, and probably only a developer can appreciate this information. But sometimes you can easily find what the problem is.

It can be useful to make a truss output during a hanging period and include it with the other files you send to Lotus Customer Support.

Another smart use of the truss tool is when a Domino process does not start, server processes included.

In this case you start the server process using truss:

```
# truss server
```

Some interesting information on why the server does not start can be found in this way.

Tip: You can also use this method for the other Domino processes. As the Notes user you can start a Domino process from a UNIX prompt. This is more complicated than starting the server because a lot of initialization steps are done by the server.

12.10.2 The sotruss utility

The **sotruss** utility executes the specified command and produces a trace of the library calls that it performs. Each line of the trace output reports what bindings are occurring between dynamic objects as each procedure call is executed.

In contrast to the **truss** command, the **-p pid** option does not exist for **sotruss**, so it is less useful than **truss**. However, it can be useful to check which shared libraries are loaded by the process at run time.

Example 12-20 is an extract from an example using the DECS process.

Example 12-20 Output from truss for the DECS process

```
# sotruss decs
```

```
decs -> libc.so.5:*_ex_register(0xfe2ed1ec, 0x0, 0x0)
decs -> libc.so.5:*_ex_register(0xff1756b0, 0xfe2f687c, 0xfe2f6884)
decs -> libc.so.5:*_ex_register(0xfe3b481c, 0xff06ae00, 0xff05d424)
decs -> libc.so.5:*_ex_register(0xff05ecc8, 0x0, 0x0)
```

Note: The **sotruss** command was introduced in Solaris 7.

12.10.3 The snoop tool

There is a useful tool on Solaris called **snoop**, which is used to capture and inspect network packets.

To see which information at the network level your Domino server is receiving, enter the command:

```
# snoop -d device -o filename
```

where device is the network interface name, usually hme0, and filename is the output file.

You can also use **snoop** to capture only the packets coming from a Notes client; for example, if the Notes client has IP address 9.95.36.113, use the command:

```
# snoop -d device -o filename 9.95.36.113
```

In this case you should see the following line in your output file, where D=1352 is the destination port of the Domino server and S=2101 is the Port number of the client:

```
51 0.24423 9.95.36.113 -> bifrost.lotus.com TCP D=1352 S=2101 Syn Seq=6314830
Len=0 Win=8192 Options=<mss 1460>
```

To read a file created by **snoop** use the -i option and redirect the output to a text file:

```
# snoop -i ./filename > filename.txt
```

Note: You must be in the root to use the **snoop** command.

The **snoop** command can provide a lot of information. For further details, see the product documentation.

12.11 Troubleshooting tips

This section describes some of the steps you can take to resolve several common problems.

12.11.1 The server does not start

If a server fails to start after a crash, it is usually because there are some resources which have not been properly cleaned up.

When the server tries to allocate these resources it meets with a failure and exits. The first step should be to run **nsd -kill** to clean up any shared memory, semaphores, or server processes which have not exited.

If this fails, there may have been a problem with **nsd -kill**, and the environment should be checked manually.

Issue the command:

```
# ps -ef | grep notes
```

where notes is the name of the Notes user. This will give a list of all the remaining processes running under that user.

The Domino application processes will have the binary path within the process name (usually /opt/lotus/notes/latest/sunspa).

Here is a example of the command with two processes which have failed to exit:

```
# ps -ef | grep notes
```

Example 12-21 Viewing processes that have failed to exit

```
notes 10130 10022 0 Mar 21 ? 0:00 /usr/dt/bin/dtscreen -mode blank
notes 10071 10069 0 Mar 21 pts/4 0:00 /bin/ksh
notes 17261 1 0 10:06:55 pts/5 0:13 /opt/lotus/notes/latest/sunspa/server
notes 17350 1 0 11:35:53 pts/1 0:00 grep notes
```

To clean up these processes issue a **kill -9** against the Domino server processes remaining:

```
# kill -9 17261
```

After the process table is cleared you also need to check the IPCS table to ensure all the semaphores that the processes had allocated are cleaned up.

In R5 on Solaris, Domino no longer utilizes System V shared memory but instead uses mmap files which reside in /tmp.

Issue the command:

```
# ipcs -sa
IPC status from <running system> as of Aug 28 11:43:47 EST 2001
T  ID   KEY   MODE  OWNER  GROUP  CREATOR  CGROUP  NSEMS  OTIME  CTIME
Semaphores:
```

If there are any Semaphores listed as being owned by the Notes user, they should be removed with the command:

```
# ipcrm -s ID
```

where ID is the Semaphore ID listed in the table.

Finally, the /tmp filesystem should be checked to see if any mmap files still exist, using the command shown in Example 12-22.

Example 12-22 Checking the /tmp filesystem for remaining mmap files

```
# ls -la /tmp
total 58786
drwxrwxrwt  9 sys  sys    2793 Mar 28 11:49 .
drwxr-xr-x 35 root  root   1024 Mar 20 13:09 ..
-rw-r--  1 notes notes  3785776 Mar 28 11:49 .NOTESMEM_please_do_not_remove.f802a800
-rw-r--  1 notes notes    0 Mar 28 11:49 .NOTESMEM_please_do_not_remove.f802a800.LCK
-rw-r--  1 notes notes  8388608 Mar 28 11:49 .NOTESMEM_please_do_not_remove.f802a801
-rw-r--  1 notes notes  8388608 Mar 28 11:49 .NOTESMEM_please_do_not_remove.f802a802
-rw-r--  1 notes notes  8388608 Mar 28 11:49 .NOTESMEM_please_do_not_remove.f802a803
drwxrwxr-x  2 root  root    176 Mar 15 15:46 .X11-pipe
drwxrwxr-x  2 root  root    176 Mar 15 15:46 .X11-unix
```

All of the files listed here which begin with “.NOTESMEM_please_do_not_remove” are mmap files which should be deleted if the server is down and the files still exist.

Issue the command:

```
#rm /tmp/.NOTESMEM_please_do_not_remove*
#ls -la /tmp
total 48
drwxrwxrwt  9 sys  sys    2793 Mar 28 11:49 .
drwxr-xr-x 35 root  root   1024 Mar 20 13:09 ..
drwxrwxr-x  2 root  root    176 Mar 15 15:46 .X11-pipe
drwxrwxr-x  2 root  root    176 Mar 15 15:46 .X11-unix
```

With the resources cleared out the server should now be able to start.

12.11.2 Server crashes or hangs immediately after startup

When a server crashes immediately or shortly after startup, the simplest way to troubleshoot the problem is to comment out all of the server tasks from the Notes.ini and see if the server will start.

Comment out the Servertasks= line by placing a semicolon (;) in front of it.

```
;Servertasks=Replica,Router,Update,Stats,AMgr,Adminp,Sched,Ca1Conn,Event,http
```

If the server restarts properly at this point, start loading the server tasks one by one from the server console to determine which one is causing the problem.

There are some common databases which could become corrupt and cause the server to crash. The following databases can be removed from the operating system while the server is down. The server will recreate them when it is restarted.

- ▶ mail.box
- ▶ log.nsf
- ▶ admin4.nsf
- ▶ busytime.nsf
- ▶ catalog.nsf
- ▶ statrep.nsf
- ▶ events4.nsf

The mail.box database can be opened as if it were an .nsf file and any unsent messages from the original database can be cut and pasted into the recreated mail.box.

Note: If it was a corrupt message which brought the server down in the first place, then copying and pasting the old message into the new mail.box may cause the crash to reoccur. Therefore, it is recommended that you cut and paste only a few messages at a time and wait for the server to process them before cutting and pasting the next several messages.

To avoid data loss, it is highly recommended that you create a temporary directory and move any suspect databases to that directory while troubleshooting the issue, rather than removing them with the **rm** command.

It is possible for the Domino Directory database to become corrupted, which may cause the server to crash. You can rule this possibility out by pulling an OS copy of the database, either from the central server or from a server which has a replica copy of the names.nsf, and restarting the server.

12.11.3 Semaphore timeouts

In a multitasking environment there is often a requirement to synchronize the execution of various tasks or ensure that one process has been completed before another begins. This requirement is facilitated by the use of a software switch known as a *semaphore* or a *flag*. The switch works in much the same way a railway signal would: only allowing one train on the track at a time. A semaphore timeout occurs when the railway signal has been set in one state too long, maybe because the train has broken down.

Semaphore timeouts in Domino

An example of a semaphore timeout in Notes/Domino occurs when the indexer needs to completely rebuild an index. It locks a semaphore so that other tasks cannot use the index until it is rebuilt. If a user task tries to open that index while it is being rebuilt, it will have to wait for the indexer to finish the rebuild and then unlock the semaphore. As a result, the user task is stuck until that semaphore is unlocked. While it is stuck waiting for the semaphore, the user task keeps track of how long it has been waiting. If it is stuck for more than 30 seconds, this is considered a semaphore timeout, and in debug mode a message will be logged to the console. The task will continue to wait for the semaphore, timing out every 30 seconds, until the semaphore is unlocked or the task is ended. For most operations, a task might only wait a few microseconds and, therefore, not time-out. But with a complicated view on a large database, the task may have to wait several minutes for the index semaphore.

If an important semaphore is locked by a task and is never unlocked, all tasks can be stopped waiting for that semaphore. This can happen in several different ways. The most common is where a task locks the semaphore and then crashes. It can also happen if a task locks the semaphore and then goes into an endless loop or gets an error and forgets to unlock it.

Semaphore deadlock can occur when two tasks try to lock two different semaphores in a different order. For example, Task A locks Semaphore 1 and then tries to lock Semaphore 2. In the meantime, Task B has already locked Semaphore 2 and is now trying to lock Semaphore 1. Task A is stuck waiting for Semaphore 2 and Task B is waiting for Semaphore 1—deadlock.

When you receive semaphore timeout messages, the messages are usually the result of one of the following:

- ▶ A heavy load on the server is causing processes to be delayed from releasing semaphores.
- ▶ A process has crashed while holding a semaphore, causing other processes to block when trying to acquire the semaphore.

- ▶ **Deadly embrace:** semaphore contention where two tasks are waiting on each other and neither task is able to break the loop. In the simplest case, thread A is trying to get a semaphore which is owned by B, while B is trying to get a different semaphore which is owned by A. More complex combinations are also possible: A wants a semaphore owned by B, which wants a semaphore owned by C, which wants a semaphore owned by A, and so forth.
- ▶ If a process fails to set a semaphore during execution, another process dependent on the semaphore will be blocked awaiting the semaphore.

12.11.4 How to set up semaphore debug

Note: Before enabling any debug parameters, we recommend that you discuss them with a Lotus Notes Support Analyst since there may be issues surrounding their use, or special precautions that must be considered. For instance, the debug parameters may require a large amount of disk space, dependent upon when the server encounters problems; the longer the server stays up, the larger the debug files will be. These files can grow large enough to cause disk space shortages.

Set debug by adding the following two lines to your Notes.ini file with the **set config** command from the Administration client:

```
Debug_Capture_Timeout=1
Debug_Show_Timeout=1
```

The first setting, `Debug_Capture_Timeout`, causes the semaphores to be written to SEMDEBUG.TXT while the other, `Debug_Show_Timeout`, causes the output to appear on the server console.

A script is provided for monitoring. To track output from **vmstat**, **iostat**, or **netstat** every 30 seconds for 2 hours and write the output to a file in the /tmp directory, create a script with the following lines:

```
%vmstat 30 240 >/tmp/vmstatmdd.log &
    (& will put this command in the background)
%iostat 30 240 >/tmp/iostatmdd.log
%netstat 30 240 >/tmp/netstatmdd.log
```




Lotus iNotes Web Access

This chapter touches lightly on the new product shipping with Domino R5.0.8, iNotes. iNotes is a new way of accessing mail on a Domino server from either a browser, or a Microsoft Outlook client. Most of iNotes' functionality is stored in a user's mail file. Therefore, much of the configuration required on the Solaris platform is the same as on the other platforms supported by Domino.

For more information on iNotes Web Access, see the IBM Redbook *iNotes Web Access*, SG24-6518.

13.1 Lotus iNotes Web Access

iNotes is comprised of two components, one for accessing mail from a Web browser and the other from a Microsoft Outlook client. In this chapter we only look at the iNotes Web Access component.

13.1.1 High-level overview

iNotes Web Access is a powerful way to access Domino's core messaging, collaboration and PIM functions through a Web browser while allowing users to work both online and offline. iNotes Web Access is a completely rearchitected Web-based client that takes the functionality of the old Lotus WebMail client much further. Companies can let current Notes users access Domino-based messaging, PIM, and collaborative services using iNotes Web Access from a Web browser, as well as giving features to new users without requiring them to run the Notes client. In addition, a company can use the iNotes Web Access technology, to reach customers and business partners without requiring them to run anything but a Web browser on their PCs.

Application Service Providers (ASPs) and Internet Service Providers (ISPs) can also use this new Web client. Using iNotes Web Access, they can give small- to medium-sized businesses messaging, collaboration, and PIM features, including the ability to work both online and offline. When used with Domino Off-line Services, which allows users to work with Domino Web applications off line, ASPs and ISPs can give customers access to Domino-based intranet applications.

iNotes Web Access builds upon the previous generation WebMail template in its use of Domino Off-line Services. With iNotes Web Access, users can work from a disconnected Web session to manage e-mail messages, contacts, calendars, and to-do items. iNotes Web Access can also work in conjunction with the Notes client or independently of the Notes client while offering many Notes core messaging features. Users can move between iNotes Web Access and the Notes client, using Notes when at their desks and using iNotes Web Access when they only have access to a Web browser, all of which is supported by a single Domino infrastructure.

iNotes Web Access provides users with almost universal access to their Notes mail and PIM functions. They can access that information from any location, such as an Internet Cafe, an Internet kiosk, or another user's PC. iNotes Web Access is also well-suited for users that routinely share a PC.

For administrators, iNotes Web Access provides a simple client that is easy and cost-effective to manage and deploy, all from within the same Domino infrastructure that you may already manage. The thin-client and server-based deployment model, as well as the absence of training requirements, will allow companies to get users up and running quickly.

13.1.2 Design goals

Since iNotes Web Access is the next generation of the WebMail client, it was completely redesigned to take advantage of the latest Internet technologies. Based on XML, DOM level 2, DHTML, and XSL, iNotes Web Access uses these technologies to deliver an advanced Web client experience. Additionally, the new Web client can easily be integrated into sites using the current Notes R5 client, so that users can enjoy interoperability between the two clients.

iNotes Web Access utilizes sophisticated JavaScript components, providing a very rich user interface within a Web browser. These components include date, time and duration controls, an outline control, tabs within forms that don't require data moving back and forth between server and client, hover and right-click menus, and a sophisticated Notes-like view component. While most Web applications break up large lists into multiple pieces or pages, this new virtual list view component allows users to view all the documents on a single page, navigable through a virtual scroll bar. Portions of the view are retrieved from the server in an XML format and incorporated into the virtual list as needed.

As with any product revision, iNotes Web Access incorporates feedback from Lotus WebMail customers that improves the WebMail user experience. Focused on ease-of-use, performance and presentation of information, these improvements make it easier for users to accomplish specific tasks, as well as simplifying user options within the Web client.

iNotes Web Access is a Domino-based application that requires minimal effort to deploy. This first version includes support for Windows NT 4.0, Windows 2000, Solaris, AIX and AS/400. A version for S/390 is currently scheduled to ship soon after initial release, and other platforms are scheduled to be supported in the second feature release. On the client, the first version supports Internet Explorer

Offline support

iNotes Web Access overcomes one of the largest problems facing large customers wanting to give their users browser-based access to their applications—the ability to seamlessly work offline. For Notes R5 users, running the Web client gives them an alternate means to replicate (or synchronize) their mail database and to work offline using a browser instead of the Notes client.

When using iNotes Web Access for the first time, users will notice a “Work Offline” link in the upper right corner of the browser window. Selecting that link will bring them to a Web page that offers options for downloading the offline components, including the iNotes Synch Manager, as well as the contents of their NSF (Notes) files. Synchronizing content between the local system and the server is then as convenient as two mouse clicks. Synchronizing delivers a comparable experience to replication using the Notes client.

User interface

Another goal with iNotes Web Access was to present the user with an intuitive Web interface, helping corporate customers to offer a powerful new tool with little retraining for users.

The iNotes Web Access user interface provides two means of moving between components of the Web client, as well as performing actions. The Task Bar is the higher menu bar, providing users with access to any of the six components on the Web client including mail, calendar, to-do items, contacts, notebook, and the Welcome Page. (In the R5 Notes client, access to these functions is gained via the icons along the left of the screen.) These roll-down menus allow users to move directly to the part of the application they desire with a single click, such as pulling up a 5-day calendar view while working in the mail inbox.

The Action Bar provides users with the contextual tools to work within the application components of the iNotes Web Access client. The “new” roll-down menu is always available to users regardless of what part of the application they are working in, including open messages and calendar items. This menu gives them the ability to create a new message, calendar entry, to-do item, contact, notebook page, or folder—with a single click. Each component of the client, mail, calendar, contacts, etc., has additional component-specific Action Bar menus. Users can also access some contextual menu items by clicking the right mouse button.

The Welcome Page is a user-customizable page that presents a main point of entry into the components of the Notes client, such as mail and contacts, as well as Web pages. Users can define rules for prioritizing messages displayed on the Welcome Page, add links to important Web sites or online documents, and check who is online (if the optional Sametime collaboration server is running).

iNotes Web Access includes a number of new user interface conventions that greatly enhance the experience of users moving from WebMail. The product opens a new window whenever a user opens a mail message, calendar entry, to-do item, contact, or notebook entry. This allows the user to more quickly take follow-up action to an event. In the instance a user receives a mail message that requires scheduling a group meeting, sending a memo, and following up on

multiple action items, having the flexibility to create those new items while viewing the original message will make the user more productive. iNotes Web Access also includes a virtual scroll bar for easier viewing of large lists and navigation within its components.

Server-based administration

Since iNotes Web Access runs on Domino/Notes R5, administrators will find they have at their disposal the same strong management tools they are accustomed to with the Web client.

Administrators can set mail quotas and enable archiving as they would with a normal Notes client.

For the user, advanced features such as delegating mail, calendar, and contacts are available, as is the ability to set time format display options and reset the HTTP password. Mail that has been read in Notes appears as read in iNotes Web Access and vice versa.

New features for WebMail users

iNotes Web Access includes a wide range of new features that improve the user experience for WebMail users as well as a few enhancements that are not currently in the Windows-based Notes client. The following is a chart of features and enhancements that improve the user experience for WebMail users specifically.

13.2 Installation

The following section is designed to assist you during setup of iNotes Web Access.

13.2.1 To install Domino server

As previously mentioned, iNotes Web Access is mostly contained within a user's mail file and therefore configuration and setup are relatively easy. In addition to the user's mail file, there is one other database, forms5.nsf, that is accessed by all iNotes users and where common JavaScript and HTML code is stored so that they can be cached by the server and the browser.

Use the following steps to install the Domino server.

1. Run the Domino setup program as described in 3.2, "Install Domino server code" on page 46.

2. When you reach the screen shown in Figure 3-4 on page 48, select the Domino Application Server or Enterprise server.
3. When you reach the screen shown in Figure 3-21 on page 60, ensure that DOLS (Domino Off-line Services) and iNotes Web Access code is installed. Leave all default selections checked. Select Domino Server Program Files, click **Change**. **DOLS Download** and **iNotes Web Access** should be selected.
4. Verify that the Organization Identity, New Server Identity, Administrator's Identity, Network Options and Communications Port Options settings are correct. Make any necessary changes. Remember to write down passwords. We recommend you reset these passwords to a more meaningful value.
5. Click Finish. Domino should now be installed.

Important: On the final screen, set the Access Control List Entry by selecting **Set Access Control List Entry**. Select **Add a group button** and in the field "Please enter the name you want added to the access control lists:" enter the "Administrator" group name. Also check the "Also add 'Anonymous' with No Access" box.

6. Click **Exit Configuration**.
7. Launch the Domino Server as described in Section 3.5, "Starting the Domino server" on page 81.
8. Run the Domino Administrator client from a Windows PC or use the Web Administration client to create and configure users. We will use the Domino Administrator client in the remaining steps.

13.2.2 iNotes Web Access configuration

Much of the iNotes Web Access configuration is achieved by customizing the settings in the server's Configuration Document. To view these settings, select the Configuration view in the Domino Directory and then select the iNotes server you wish to configure. Figure 13-1 on page 361 shows the options available to you in this document.

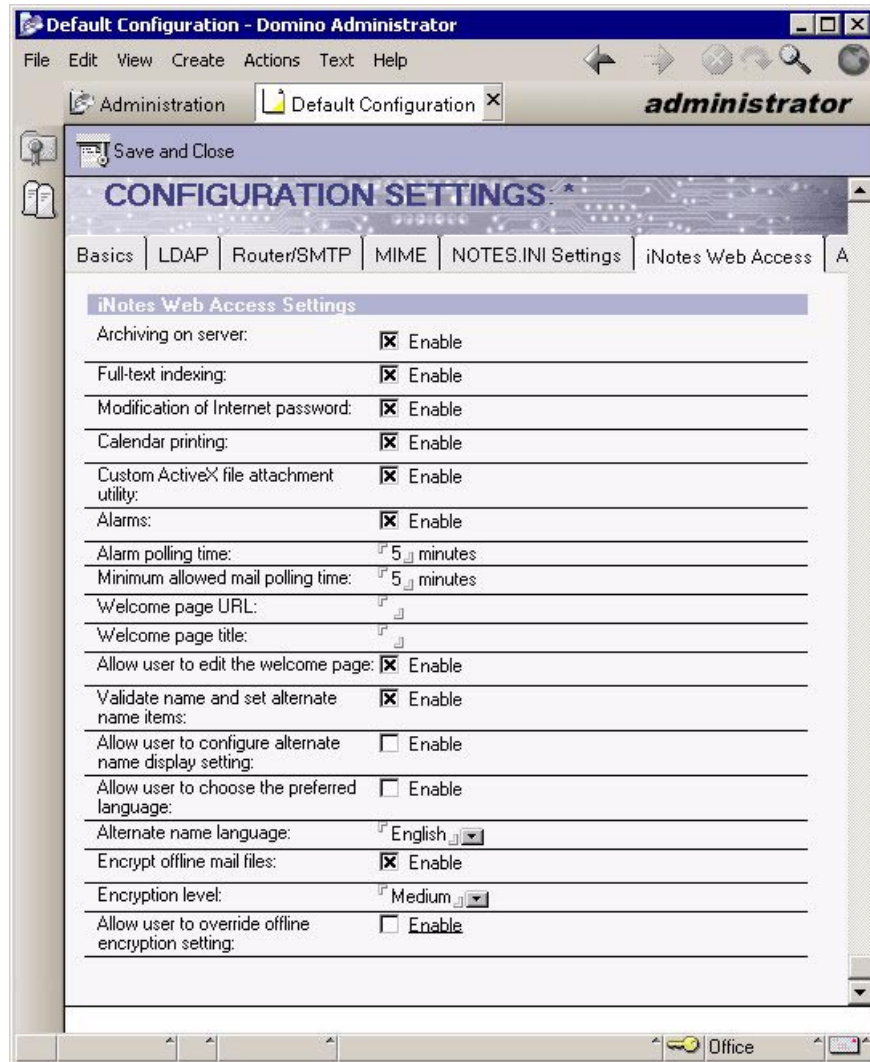


Figure 13-1 iNotes Web Access configuration tab

13.2.3 To create user accounts for use with iNotes Web Access

Once the Lotus Domino Administrator client has launched, use the following steps to create additional user accounts:

1. Select the People & Groups tab.
2. On the far right, click **People**.
3. Select **Register**.

4. Close the Certifier ID window (ID should be in the data directory).
5. Ignore the warning message regarding Certifier recovery information.
6. Under the Register Person window, select **Advanced**.
7. Enter the user's first and last name and password. You can also set this password to be the Internet password by selecting the checkbox. The Internet password is required for iNotes Web Access. Make note of the Short Name, that is, the name used to log on using iNotes Web Access.
8. Make any other additional selections you require and click **Add person**.
9. Click **Register** at the bottom of the page.
10. Click **Done**.

13.2.4 To fully enable iNotes Web Access on the browser client

There are no true "installation" steps for the browser client. An end user's use of the product will cause him to be prompted to install the upload/download component, as well as the iNotes Synch Manager application if the user chooses to go offline. To access iNotes, the user should point their browser to the following URL:

`http://servername/mail/username.nsf`

If the browser window prompts the user to download various controls, they should select **Yes**.

Note: To convert regular Notes mail databases to iNotes mail databases you can use the Domino server **convert** command.



Using Notes C API to make your own backup tool

This appendix shows how to make your own backup tool using the Notes C API Library for Domino R5.

Note: The information contained in the following Appendix was written using Domino R5.02a and Solaris 7. It was not tested again for Domino R5.0.8 and Solaris 8 but is included here for information.

The backup and recovery functionality of the Lotus C API for Domino and Notes allows the media recovery of databases under Domino R5. This API functionality covers capturing a full point-in-time backup of a database, archiving transaction log file extents, and finally, recovering a database by applying its logged transactions to its recoverable backup file.

We compiled and tested the samples shipped with the Lotus C API toolkit, under the directory samples/admin/backup. We used the Lotus C API toolkit version 5.02.

The sample in the Lotus distribution is for the Windows NT platform; we made some modifications in the sources to port the application to the Solaris platform. You have to get the `print_api_error()` function, used to print out the API error messages, from the `win32io.c` file and add it to the `dbbackup.c` and `dbrecs.c` files.

The two reported examples are:

- ▶ DBBACKUP - Create a backup file of a database
- ▶ DBRECS - Archive/check logs or recover/restore a logged database from a backup file

The command line of dbbackup is:

dbbackup <database filename> <output filename>

where:

- ▶ <database filename> is the filename of the Domino database to backup
- ▶ <output filename> is the filename of the backup file created

The command line of dbrecs is:

dbrecs <option> [input file] [restore db]

where:

- ▶ <option> is the action to be performed, from among the following:
 - ARCHIVE - Archive system logs (Input file *not* required)
 - CHECK - Check a database or backup file (Input file *required*)
 - RECOVER - Recover a backup file (Input file *required*)
 - RESTORE - Restore a database (Input file & restore db *required*)
- ▶ [input file] is the path to a backup file in the Domino data directory
- ▶ [restore db] is the path to the database in the Domino data directory to be restored.

To compile the samples we used the GNU C compiler gcc version 2.95.2. We downloaded it from the Solaris free software Web site at www.sunfreeware.com.

These programs are intended to illustrate the practical usage of the backup, media recovery, and archiving functions of the Backup and Recovery API for Domino R5.

Transactional system logging *must* be enabled in order to implement the recovery of databases using the Backup and Recovery API.

This is the modified makefile that we used on Solaris with the gcc compiler:

```
# makefile for Notes API sample program dbbackup
#           Solaris 2 SPARC Edition
# set TARGET to the name of the executable to create
TARGET = dbbackup
```



```

# set SOURCES to the list of C source files in this program
SOURCES = $(TARGET).c
# set HEADERS to the list of C include files in this program
HEADERS =
# set OBJECTS to the list of object files that must be linked
OBJECTS = $(TARGET).o
# Link this program with the bootstrap code notes0.o because
# this program is structured as a NotesMain.
#BOOTOBS = $(LOTUS)/notesapi/lib/sol_2x/notes0.o
# CC defines the compiler. Set to "cc"
CC = gcc
# Set CCOPTS - the compiler options.
CCOPTS = -c
# You may use -g flag for debugging:
#CCOPTS = -c -g
# set NOTESDIR to specify where to search for the Notes library file
NOTESDIR = $(Notes_ExecDirectory)
# Set LINKOPTS - the linker options passed to CC when linking.
# -o $(TARGET) causes compiler to create target rather than a.out
LINKOPTS = -o $(TARGET) -R $(NOTESDIR)
# Notes API header files require UNIX to be defined.
DEFINES = -DUNIX -DSOLARIS
# set INCDIR to specify where to search for include files
INCDIR = $(LOTUS)/notesapi/include
# set LIBS to list all the libraries ld should link with.
LIBS = -lnotes -lm -lnsl -lsocket -lposix4 -lc
# the executable depends on the objects.
$(TARGET): $(OBJECTS)
    $(CC) $(LINKOPTS) $(OBJECTS) $(BOOTOBS) -L$(NOTESDIR) $(LIBS)
# the object files depend on the corresponding source files
.c.o:
    $(CC) $(CCOPTS) $(DEFINES) -I$(INCDIR) $(SOURCES)

```



B

Creating a UNIX partition

The following procedures are used to create a UNIX partition on a Solaris server using the **format** and **newfs** utilities.

The **#** sign defines the Solaris account level root and the prompt displayed. Any commands will follow this prompt.

In order to use the **format** utility correctly, you should first know which disk you want to create a new partition on and how much space is available on the disk for the partition.

Solaris provides seven partitions per disk. Partition 2 is always the whole disk size and should not be used. There are a number of commands that can be used to determine which partitions are being used and to show the space allocated on the disks.

df -k

This command will provide a list of the disks that are mounted on the system, how much disk space is allotted and how much remains.

```
# df -k
```

Table B-1 Example output from df -k

File system	Kbytes	Used	Available	Capacity	Mounted on
/proc	0.00	0.00	0.00	0%	/proc
/dev/dsk/c0t0d0s0	145135.00	36737.00	93885.00	29%	/
/dev/dsk/c0t0d0s4	866575.00	578320.00	227595.00	72%	/usr
fd	0.00	0.00	0.00	0%	/dev/fd
/dev/dsk/c0t0d0s5	96391.00	6507.00	80245.00	8%	/var
/dev/dsk/c0t0d0s3	1428431.00	339679.00	1031615.00	25%	/opt
swap	3244816.00	61080.00	3183736.00	2%	/tmp
/dev/dsk/c0t1d0s3	1984903.00	239356.00	1686000.00	13%	/export/notesdata4
/dev/dsk/c0t2d0s3	1984903.00	236540.00	1688816.00	13%	/export/notesdata5
/dev/dsk/c0t2d0s4	1984903.00	239665.00	1685691.00	13%	/export/notesdata6
/dev/dsk/c0t3d0s3	1015679.00	14.00	954725.00	1%	/export/home
/dev/dsk/c0t3d0s4	1015679.00	11.00	954728.00	1%	/export/notesdata7

Prtvtoc

This command can be used to see what disk partitions have been used in relation to a specific logical disk identification.

For example, **prtvto** /dev/rdsk/c0t0d0s0 would produce the results shown in Example B-1.

Example: B-1 Output from the prtvto command

```
# prtvto /dev/rdsk/c0t0d0s0
* /dev/rdsk/c0t0d0s0 partition map
*
* Dimensions:
*   512 bytes/sector
*   135 sectors/track
*   16 tracks/cylinder
* 2160 sectors/cylinder
* 3882 cylinders
* 3880 accessible cylinders
*
* Flags:
*  1: unmountable
* 10: read-only
*
```

In the following table you have the partitions that are being used on the C0t0d0 disk. Later in this appendix we explain the definition of each logical device identification.

Partition	Tag	Flags	First Sector	Sector Count	Last Sector	Mount Directory
0.00	2.00	00	0	308,880	308,879	/
1.00	3.00	01	308,880	3073680.00	3382559.00	
2.00	5.00	00	0	8380800.00	8380799.00	
3.00	0.00	00	3382560.00	2948400.00	6330959.00	/opt
4.00	4.00	00	6330960.00	1844640.00	8175599.00	/usr
5.00	7.00	00	8175600.00	205,200	8380799.00	/var

By using the two commands you can determine which disk is currently being used and what partition is available to be configured. Once this has been determined, you can use the following steps to invoke the format command utility.

1. You must be logged in on the server as Solaris Account root, or **su** to the root level.

format

When you type this command a menu will appear showing all the disk drives that are installed on your system.

```

# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
 0. c0t0d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@3/sd@0,0
 1. c0t1d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@3/sd@1,0
 2. c0t2d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@3/sd@2,0
 3. c0t3d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@3/sd@3,0
 4. c2t0d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@4/sd@0,0
 5. c2t1d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@4/sd@1,0
 6. c2t2d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@4/sd@2,0
 7. c2t3d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@4/sd@3,0
 8. c3t0d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>
    /pci@1f,4000/scsi@4,1/sd@0,0
 9. c3t1d0 <SUN9,0G cyl 4924 alt 2 hd 27 sec 133>

```

Figure 13-2 Output from the format command

2. Select the disk that you want to create a partition on. The disk will have incremental numbers assigned to the logical address of the disk. The standard disk identification will resemble “/dev/dsk/c0t0d0s0”. The definition is as follows:

c0 - Controller 0

t0 - Target slot 0 - The device attached to the controller incremented

d0 - Disk 0 - The disk identifier

s0 - partition number on the disk (also known as slice of disk)

3. To find out which disk is identified, you can use the following command from the # prompt:

```
# ls -l /dev/dsk/c0t0d0s0
```

```
# ls -la /dev/dsk/c0t0d0s0
```

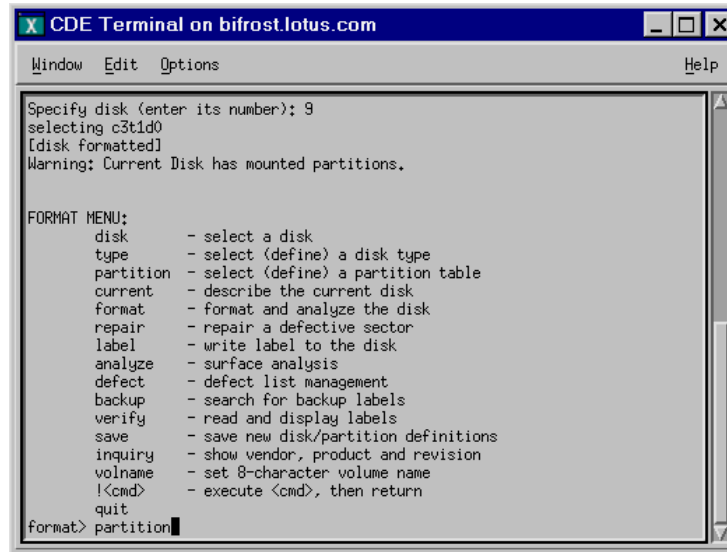
```
lrwxrwxrwx 1 root root 41 Mar 9 17:13 /dev/dsk/c0t0d0s0 ->
```

```
../../devices/pci@1f,4000/scsi@3/sd@0,0:a
```

This example shows that the device name /dev/dsk/c0t0d0s0 is linked to the physical device /devices/pci@1f,4000/scsi@3/sd@0,0 which is disk 0 from the format command.

4. Once you have decided what disk you want to partition and selected it, a second menu will appear which will give you many options on tasks to

perform on the disk, such as format, test, analyze, and so forth. We select the partition option by typing **partition** at the format prompt:
format > partition

A screenshot of a CDE Terminal window titled "CDE Terminal on bifrost.lotus.com". The window has a menu bar with "Window", "Edit", "Options", and "Help". The terminal text shows the following sequence: "Specify disk (enter its number): 9", "selecting c3t1d0", "[disk formatted]", "Warning: Current Disk has mounted partitions.", and a "FORMAT MENU:" listing various options like disk, type, partition, current, format, repair, label, analyze, defect, backup, verify, save, inquiry, volname, !<cmd>, and quit. The "format" option is highlighted, and the prompt "format> partition" is shown at the bottom.

```
CDE Terminal on bifrost.lotus.com
Window Edit Options Help

Specify disk (enter its number): 9
selecting c3t1d0
[disk formatted]
Warning: Current Disk has mounted partitions.

FORMAT MENU:
  disk      - select a disk
  type      - select (define) a disk type
  partition - select (define) a partition table
  current   - describe the current disk
  format    - format and analyze the disk
  repair    - repair a defective sector
  label     - write label to the disk
  analyze   - surface analysis
  defect    - defect list management
  backup    - search for backup labels
  verify    - read and display labels
  save      - save new disk/partition definitions
  inquiry   - show vendor, product and revision
  volname   - set 8-character volume name
  !<cmd>    - execute <cmd>, then return
  quit

format> partition
```

Figure 13-3 Formatting a partition

5. The next menu will be the partition menu with the disk table identified. Select **print** from the partition prompt:
partition > print

```

6 - change '6' partition
7 - change '7' partition
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
!<cmd> - execute <cmd>, then return
quit
partition> print
Current partition table (original):
Total disk cylinders available: 4924 + 2 (reserved cylinders)

Part   Tag      Flag    Cylinders      Size      Blocks
0 unassigned  wm       0              0      (0/0/0)      0
1 unassigned  wm       0              0      (0/0/0)      0
2 backup     wu       0 - 4923      8.43GB    (4924/0/0) 17682084
3 alternates wm       0 - 2336      4.00GB    (2337/0/0) 8392167
4 unassigned  wm       0              0      (0/0/0)      0
5 unassigned  wm       0              0      (0/0/0)      0
6 unassigned  wm       0              0      (0/0/0)      0
7 unassigned  wm       0              0      (0/0/0)      0
partition>

```

Figure 13-4 Displaying the current partition information

6. The disk settings will be defined. From this point you can configure the disk table and set the use for each section and the size.

This is the same utility you used if you chose the manual disk layout when you installed your Solaris software. The steps for creating a partition follow. For complete explanations of all the variables, consult the Solaris manuals and online documentation.

- a. Select the partition number you want to use.

Important: Never use partition 2, as this defines the overlap area of the disk which is considered the total disk area.

- b. Select the partition ID. For non-predefined OS partitions you can use **alternate** as the partition ID. Valid predefined OS partition IDs are **root**, **usr**, **swap**, etc.
- c. Select the partition permission flag: **wm** for standard disk, **wu** for swap disk.
- d. Identify the starting cylinder number to be used for this partition. If you have previously defined a partition, the last cylinder number will be displayed. Start with the next incremental number of the cylinder. In the

example begun previously, the partition 3 cylinder was from 0–2336. So you would start your cylinder number at 2337.

- e. Enter the partition size. This can be done in cylinders, blocks, megabytes, or gigabytes. Enter the size that you would like for the partition, for example, 3.0 GB.
- f. When you have finished, select **print** again from the **partition** prompt to see your results.

```

CDE Terminal on bifrost.lotus.com
Window Edit Options Help

Part  Tag  Flag  Cylinders    Size    Blocks
  4  unassigned  wm      0          0    (0/0/0)      0

Enter partition id tag[unassigned]:
Enter partition permission flags[wm]:
Enter new starting cyl[0]: 4924
'4924' is out of range.
Enter new starting cyl[0]: 2337
Enter partition size[0b, 0c, 0.00mb, 0.00gb]: 3.00gb
partition> print
Current partition table (unnamed):
Total disk cylinders available: 4924 + 2 (reserved cylinders)

Part  Tag  Flag  Cylinders    Size    Blocks
  0  unassigned  wm      0          0    (0/0/0)      0
  1  unassigned  wm      0          0    (0/0/0)      0
  2  backup      wm    0 - 4923    8.43GB   (4924/0/0) 17682084
  3  alternates  wm    0 - 2336    4.00GB   (2337/0/0) 8392167
  4  unassigned  wm  2337 - 4089  3.00GB   (1753/0/0) 6295023
  5  unassigned  wm      0          0    (0/0/0)      0
  6  unassigned  wm      0          0    (0/0/0)      0
  7  unassigned  wm      0          0    (0/0/0)      0

partition>

```

Figure 13-5 Viewing changes using the `partition-print` command

- g. After you have completed creating your partition, you will need to label the disk for the change to be accepted. At the partition prompt type the **label** command.

```
partition> label
```

```
Ready to label disk, continue? y
```

Attention: Be sure to label your partition change before exiting from the partition menu. If you do not label your disk the partition changes will be deleted when you exit the partition menu.

Important: You will not be able to label any disk that has a partition mounted, so be sure to create all the partitions before mounting the disks, or unmount any previously created partitions before you begin this process.

7. To exit from the utility, type:

```
partition > quit
```

```
format> quit
```

8. Create the filesystem to be mounted. Type the following command from the # prompt:

```
# newfs /dev/dsk/logical disk
```

Example: B-2 Creating a new file system using newfs

```
# newfs /dev/dsk/c3t1d0s4
newfs: construct a new file system /dev/rdisk/c3t1d0s4: (y/n)? y
Warning: 1 sector(s) in last cylinder unallocated
/dev/rdisk/c3t1d0s4: 6295022 sectors in 1753 cylinders of 27 tracks, 133
sectors
3073.7MB in 110 cyl groups (16 c/g, 28.05MB/g, 3392 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
32, 57632, 115232, 172832, 230432, 288032, 345632, 403232, 460832, 518432,
576032, 633632, 691232, 748832, 806432, 864032, 921632, 979232, 1036832,
1094432, 1152032, 1209632, 1267232, 1324832, 1382432, 1440032, 1497632,
1555232, 1612832, 1670432, 1728032, 1785632, 1838624, 1896224, 1953824,
2011424, 2069024, 2126624, 2184224, 2241824, 2299424, 2357024, 2414624,
2472224, 2529824, 2587424, 2645024, 2702624, 2760224, 2817824, 2875424,
2933024, 2990624, 3048224, 3105824, 3163424, 3221024, 3278624, 3336224,
3393824, 3451424, 3509024, 3566624, 3624224, 3677216, 3734816, 3792416,
3850016, 3907616, 3965216, 4022816, 4080416, 4138016, 4195616, 4253216,
4310816, 4368416, 4426016, 4483616, 4541216, 4598816, 4656416, 4714016,
4771616, 4829216, 4886816, 4944416, 5002016, 5059616, 5117216, 5174816,
5232416, 5290016, 5347616, 5405216, 5462816, 5515808, 5573408, 5631008,
5688608, 5746208, 5803808, 5861408, 5919008, 5976608, 6034208, 6091808,
6149408, 6207008, 6264608,
```

You have now created a partition that can be mounted and used.

9. Before you mount your new partition, check the partition to be sure that it was created correctly. Use the **fsck** command to do this.

```
# fsck /dev/dsk/c3t1d0s4
```

Replace the logical disk partition with the one you just created.

10. Create a mount-point for your new File System (partition) using the **mkdir** command.

```
# mkdir /export/notesdata4
```

Replace the directory shown with the directory you will be mounting the File System (partition) on.

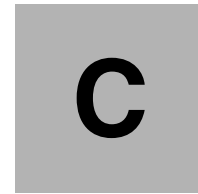
11. Mount the File System at the mount-point that you created using the **mount** command.

```
# mount /dev/dsk/c3t1d0s4 /export/notesdata4
```

Replace the logical disk and the mount-point with the one you have created.

12. Edit the /etc/vfstab file to have the File System automatically mounted when the system is rebooted. For details on the vfstab file consult your Solaris documentation.

Tip: If you are not experienced with Solaris, a good example of the mount options from the vfstab file would be to follow the parameters for the /opt File System.



Installing Domino using domsetup

The domsetup program is an application for installing a Domino server using a non-graphical environment. It is available for Domino version 4.5.x, 4.6.x, and 5.0.

It was designed to avoid the problems sometimes encountered when you do not have a graphical environment on the server or any workstation on which to export the display. With it, you can install Domino using a simple tty terminal.

Domsetup can create certificates for US or International English versions of Domino when installing a primary server. It can also set up secondary servers.

It is composed of one shell script (domsetup.sh) and a C program (domsetup) written using the Lotus Notes C API Toolkit. The script gets all the information and updates the Notes.ini, then calls the domsetup program that builds all the ID files and databases.

Domsetup is installed in the /opt/lotus directory from the CDROM. Then it prompts you to answer several questions and, when all information is provided, the Domino data directory and the Notes.ini file are upgraded. At the end you are prompted to start the server.

To use domsetup, type the command below in a temporary directory using the Notes user account:

```
$ ./domsetup.sh
```

Then just follow the online instructions, and provide a small set of parameters.

The domsetup program is not supported by Lotus. To obtain the latest version contact the program authors, Marc Riart and Eric Dolce, at their respective e-mail addresses:

mriart@lotus.com

edolce@lotus.com



D

Domino and syslog

You can send some Domino events to the syslog daemon running on Solaris. The syslogd daemon reads and forwards system messages to the appropriate log files and/or users, depending upon the priority of a message and the system facility from which it originates.

It also can be configured to receive messages from external applications, like the Domino server.

To do this you have to change the syslogd configuration file to control output from the daemon.

1. Log in as root and in the /etc/syslog.conf file add the line:

```
user.warn    /var/log/domino.log
```

Use user.* to include severities of events.

Tip: Use a tab as a separator, otherwise it does not work.

2. Specify a file name and location for the output. In this example we used /tmp/syslog.user.

Note: Normally on UNIX systems all the log files are located in the /var/log file system.

3. Create the file and command syslogd to reread its configuration file. Use the UNIX command **touch** to create an empty file and send the SIGHUP signal to syslogd using the **kill** command.

For example, if the syslogd daemon has the pid=2033:

```
touch /var/log/domino.log
```

```
Kill -HUP 2033
```

4. Switch user to notes (**su - notes**). At the Domino server console, load the event task by issuing:

```
> load event
```

The first time you load event it creates a Statistics & Events database (events4.nsf).

5. If you want event to start when the server starts, add it to the ServerTasks line in Notes.ini.

Tip: If the event task is already running, we found that you have to reload the task to make it work with the restarted syslogd.

6. Finally, you can start a Notes client, open the Statistics & Events database, and set up your Database monitors, Event monitors, and so forth.

The following is an example of the steps you would follow to monitor the ACL changes in the Domino Directory database names.nsf.

1. Open the Statistics & Events database from a Notes client.
2. Click "Select New ACL Change Monitor," as shown in Figure D-1 on page 381.

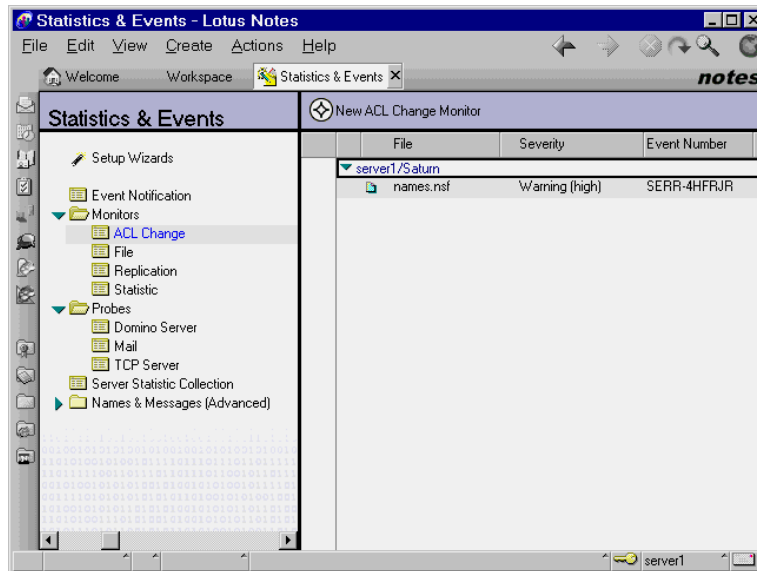


Figure D-1 ACL change monitor

3. In the Other tab, click “Create a new notification profile for this event”

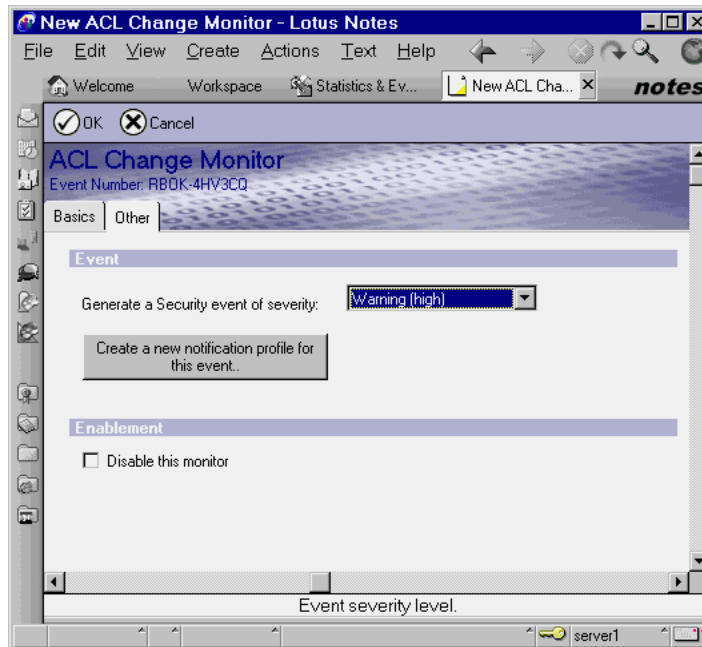


Figure D-2 Selecting a security event level for the ACL monitor

4. Select the severity from the list:

- Fatal
- Failure
- Warning (high)
- Warning (low)
- Normal

We chose user.warn in the /ect/syslog.conf file, so we chose Warning.

5. The Event Notification Wizard dialog box will appear, as shown in Figure D-3.



Figure D-3 The Event Notification Wizard dialog box

Click Next to continue.

6. Select the Log to UNIX System Log option. (See Figure D-4 on page 383.)



Figure D-4 Selecting the notification method for the wizard

7. Restart the event task with the Domino server console command:
> tell event restart
8. Change the ACL by adding a new user to the ACL list for names.nsf and you will get an entry in /var/log/domino.log file like this:
Mar 28 17:01:41 bifrost event[2092]: The ACL in notefile names.nsf has been changed by Red Book/Saturn. [SERR-4HFRJR]



Example TCP port Notes.ini settings

The following examples offer insight into how to effectively configure and tune the network connectivity aspects of your Domino servers. We have included Notes.ini network settings, a view on what the server's listener ports would look like from the Domino server's console, as well as how the server doc Notes ports need to be listed for the Notes name service to be effective, and finally, how to effectively use the different protocol name resolve services.

Basic Notes.ini settings

Standard server - single or multiple IP addresses

Standard server with single or multiple IP addresses and N-RPC port globally bound to all IP addresses present on system.

If one has only one Domino server present on the system, by default this is how it is configured. If the system has more than one IP address, each IP address's TCP port 1352 is bound to this server. If you attempt to add a second server to the system you will have a port bind conflict. In addition, you can have unexpected connectivity problems—either inbound or outbound—due to parallel pathing due to a dynamic route expiring or a default route not being the correct route path. If you have other applications (i.e. tape backup, etc.) using other interfaces, you will also likely need to specifically bind the Domino server ports.

Generally, you will want to bind the NRPC port to a single given IP address whenever possible.

```
Ports=TCPIP
tcpip=TCP, 0, 15, 0
```

Servers console doing a Show Port TCPIP

Table E-1 Example Show Port output

Notes Session	Local Address	Foreign Address
8 hex digit value	*:1352	*.* .

Multiple IP addresses on the server system

Multiple IP addresses on the server system and multi-homing/multi-netting prerequisites.

Independent of the Domino server's TCP port configurations are the requirements of the TCP/IP protocol stack and its settings. These include: IP addresses, IP address masks, the default router (gateway), and if needed, static routes or dynamic route services (RIP2 or OSPF services). Also included are the name resolve services (Host & DNS entries as required) and the resolve order on the server system. They all must be correctly configured.

There are basically two configuration designs used with Domino servers:

1. Partitioning the system with multiple Domino servers, in which one binds discreet NIC IP addresses on a one-to-one basis (multi-homed) or shares a given NIC which has multiple IP addresses defined (Multi-netted). Each IP address would be part of the same IP network (net and subnet).

2. A given Domino server needs access to two different IP networks (net or subnet). Either to isolate different groups of users pathways to the Domino server (Intra Vs Inter), or Proxied pathway (Notes Pass-Through) to Authenticate inbound connections, or to segregate traffic flows between Domino servers or services (SMTP Gateway).

IP routing issues

If you intend to use different IP networks for a given partitioned server (Design 1) you may encounter both name resolve issues as well as IP route pathing problems. These can occur if the Domino server needs to access other Domino servers which are not within the same IP network or subnet, or if the default route is associated with the other IP network. You will need at least static route entries within the server system.

With Domino servers which are accessing two different networks or subnets (Design 2) and the other Domino server being accessed is in parallel directly or indirectly besides the Domino servers NRPC port ordering, name resolve and IP route services need to be correctly set other wise you may not get the expected reaction. You may need static route entries within the server system.

With either Notes clients or Domino servers accessing a Domino server across a Network Address Translation router (NAT), depending on the NAT unit and your routers settings, you may need a static route to guide the return flow to the NAT router.

Consider the number of remote systems this Domino server system needs to access by IP network and the next hop to this IP network. Set the default Router (gateway) of the system to the router of the majority of your targets (i.e. Internet), then configure static routes for your smaller group of targets (i.e. Intranet).

IP name resolve direction

For the most part, one wants to set resolve order to use Host file, then DNS. That way a private path (server to server) can be leveraged via the host file, while other servers or clients can access the public interface via the DNS.

One can also leverage the DNS hierarchy by placing the Domino server's IP domain name lookup scope to a private IP domain (i.e. dom-wan.ibm.com). That way the server system can resolve out a private IP address were as the users and other servers are members of the root domain (ibm.com). Then have two listings with the same server name (<servers name>.dom-wan.ibm.com for the private interface address and <servers name>.ibm.com) for the public interface address. This is very effective for private WAN access paths for the Domino servers. The Domino servers which are in this lower IP domain or other subdomain will also find any server within the root domain as the lookup process

will strip the left-most domain name and probe the next one down until it has tried all of the domains. If you need to do lookups downstream, you can leverage a suffix search order entry within the TCP/IP stack or leverage the Notes name service if it is available.

With either a multi-homed or multi-netted host that accesses two or more different IP networks you need to consider which network's DNS system the server system will look to for remote target name resolve. In most cases this DNS system (or systems) should be within the IP network or subnetwork the default gateway is set to.

Consider the number of remote IP domain hosts that need to be resolved (i.e. Internet versus intranet). Also think about the network path and its dependability to the DNS system as well as the DNS system's load. You may need to enhance your DNS system, as well as improve the network to it and/or to your ISP's DNS system.

Multi-homed server

Multi-homed server with dual NIC, IP address per NIC (N-RPC ports specifically bound)

In this example we have a single server which has two NRPC ports. In this case the ordering of port names in the PORTS= entry becomes important as the server will bias this port for all outbound connections based on the Notes named network the targeted server is present on (in the case of mail and pass-through services). When using connection documents, the port offering is controlled based on the server document's listed order. Finally, the user's lookup using Notes name services will be controlled by the server document's list order. This is were, in most cases, the ordering of the INI and the server doc are opposite of each other.

Because of IP routing rules, each interface needs to be in a different IP network (or subnet).

```
Ports=TCPIP1,TCPIP2
TCPIP1=TCP, 0, 15, 0
TCPIP1_TcpIpAddress=0,10.1.1.1:1352
TCPIP2=TCP, 0, 15, 0
TCPIP2_TcpIpAddress=0,103.0.0.1:1352
```


Table E-2 Example Show Port TCPIP1t

Notes Session	Local Address	Foreign Address
8 hex digit value	10.1.1.1:1352	*.* .

Table E-3 Example Show Port TCPIP2

Notes Session	Local Address	Foreign Address
8 hex digit value	103.0.0.1:1352	*.* .

Note: The IP networks are different, otherwise you don't have isolation of data flows.

R5.0.x Internet TCP settings

Internet TCP ports are globally bound to all IP addresses when an N-RPC port is not specifically bound (the default).

By default, Internet TCP ports on partitioned servers will bind to assigned IP addresses set by N-RPC TCP port bind.

Also by default, Internet ports on multi-homed servers will bind to the first N-RPC TCP port. They can be specifically bound to different IP addresses with INI settings as needed.

The TCP Port number being used by a given service is controlled in the given server's Server document. Generally, they are left at the assigned values (as seen in these examples).

Multi-homed server port settings with Internet services

Here we are using the binding to the NRPC port method. If you use the second method you will not see the listeners or the connections in the server console. While it may be beneficial not having an exposed NRPC port on a given IP address, for the most part it is not a security problem since one can add an additional setting so all connections are denied over it.

```
Ports=TCPIP1, TCPIP2
TCPIP1=TCP, 0, 15, 0
TCPIP1_TcpIpAddress=0,10.1.1.1:1352
TCPIP2=TCP, 0, 15, 0
TCPIP2_TcpIpAddress=0,130.2.2.2:1352
SMTPNotesPort=TCPIP2
NNTPNotesPort=TCPIP2
LDAPNotesPort =TCPIP2
```

IMAPNotesPort =TCPIP2
POP3NotesPort =TCPIP2

Note: TCPIP2 port is required in this configuration

Table E-4 Example Show Port TCPIP1

Notes Session	Local Address	Foreign Address
8 hex digit value	10.1.1.1:1352	*.* .

Table E-5 Example Show Port TCPIP2

Notes Session	Local Address	Foreign Address
8 hex digit value	103.2.2.2:1352	*.* .
8 hex digit value	103.2.2.2:25	*.* .
8 hex digit value	103.2.2.2:119	*.* .
8 hex digit value	103.2.2.2:389	*.* .
8 hex digit value	103.2.2.2:143	*.* .
8 hex digit value	103.2.2.2:110	*.* .

R4.6x Internet settings

As a reference point, here are the R4.6.x Internet TCP settings. Internet TCP ports are globally bound to all IP addresses by default. They can be specifically bound to different IP addresses with INI settings as needed.

Some of the TCP Port numbers are controlled within the given Server document. The SMTP service is defined within the INI as shown here.

Example: E-1 R4.6 Multi-homed servers port settings with I-net services

```
Ports=TCPIP1, TCPIP2
TCPIP1=TCP, 0, 15, 0
TCPIP1_TcpIpAddress=0,10.1.1.1:1352
TCPIP2=TCP, 0, 15, 0
TCPIP2_TcpIpAddress=0,130.2.2.2:1352
SMTPMTA_IPAddr=130.2.2.119
SMTP_IPPort=25
LDAP_IPAddress=130.2.2.119:389
NNTP_IPAddress=130.2.2.119:119
IMAP_IPAddress=130.2.2.119:143
POP3_IPAddress=130.2.2.119:110
```

Note: Unlike this example, in R5 the Internet ports are not viewable within the network console

Partitioned server - single NIC

Partitioned server with a single NIC, multiple IP addresses or Multiple NICs within the same IP network.

ServerA

```
Ports=TCPIP
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,10.1.1.1:1352
```

ServerB

```
Ports=TCPIP
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,10.1.1.2:1352
```

ServerC

```
Ports=TCPIP
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,10.1.1.3:1352
```

Note: The IP nets can be the same since each server process is independent.

Partitioned servers - dual NIC

Partitioned server with dual NIC, multiple IP addresses and IP networks or subnetworks.

In this example we are using two subnets. The IP address range being used here is a Class A (private and non-routable on the Internet), and the address mask must be Class B (255.255.0.0) because the two NICs must be supporting different IP nets (10.1.x.x vs. 10.2.x.x). Each server has an NRPC port tied to each of these IP nets.

ServerA

```
Ports=TCPIP1,TCPIP2
TCPIP1=TCP, 0, 15, 0
TCPIP1_TcpIpAddress=0,10.1.1.1:1352
TCPIP2=TCP, 0, 15, 0
TCPIP2_TcpIpAddress=0,10.2.2.1:1352
```

ServerB

```
Ports=TCPIP1,TCPIP2
TCPIP1=TCP, 0, 15, 0
TCPIP1_TcpIpAddress=0,10.1.1.2:1352
TCPIP2=TCP, 0, 15, 0
TCPIP2_TcpIpAddress=0,10.2.2.2:1352
```

ServerC

```
Ports=TCPIP1,TCPIP2
TCPIP1=TCP, 0, 15, 0
TCPIP1_TcpIpAddress=0,10.1.1.3:1352
TCPIP2=TCP, 0, 15, 0
TCPIP2_TcpIpAddress=0,10.2.2.3:1352
```

Partitioned servers - single NIC and loopback

Partitioned servers with a single NIC, multiple IP addresses with loopback using port mapper.

The TCPIP ports' addresses could be different IP nets on each server since each server process is using an independent IP address. The TCPLB port is leveraging the Base Server's port mapper off of the standard loop back address. Each of the server's IP host names needs to be set up as an alias in the system's Host file off of the loop back port address.

For the most part this design is not needed in Solaris as the TCP/IP stack will route the flow directly back at the TCP layer. This method is mostly beneficial when you need to account for the number of kb of data being sent between the partitioned servers.

Base Server (mapper only)

```
Ports=TCPLB
TCPIP_TcpIpAddress=0,127.0.0.1:1352
TCPIP_PortMapping00=CN=BaseServer/0=ACME,127.0.0.1:1352
TCPIP_PortMapping01=CN=ServerA/0=ACME,127.0.0.1:10001
TCPIP_PortMapping02=CN=ServerB/0=ACME,127.0.0.1:10002
TCPIP_PortMapping03=CN=ServerC/0=ACME,127.0.0.1:10003
```

ServerA

```
Ports=TCPLB,TCPIP
TCPLB_TcpIpAddress=0,127.0.0.1:10001
TCPLB_PortMapping01=CN=Server1/0=ACME,127.0.0.1:10001
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,10.1.1.1:1352
```

ServerB

```
Ports=TCPLB,TCPIP
TCPLB_TcpIpAddress=0,127.0.0.1:10002
TCPLB_PortMapping02=CN=Server2/O=ACME,127.0.0.1:10002
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,10.1.1.2:1352
```

ServerC

```
Ports=TCPLB,TCPIP
TCPLB_TcpIpAddress=0,127.0.0.1:10003
TCPLB_PortMapping03=CN=Server1/O=ACME,127.0.0.1:10003
TCPIP=TCP, 0, 15, 0
TCPIP_TcpIpAddress=0,10.1.1.3:1352
```




Example script to start and shut down a Domino server

The following script can be used to start and shut down the Domino server(s).

This script is available for downloading from the ITSO website, see “Locating the Web material” on page 419.

Example: F-1 Domino startup script

```
#!/bin/sh
#
# Copyright (c) 1999-2001 by Sun Professional Services &
# Technology Product & Consulting
# Additional Parts (c) 1999-2001 by PRS GmbH
#
# ident"@(#)lotus3.801/09/10 TPC Sun Microsystems"
#
# Authors & Maintainers:
#
# Klaus.Ziegler@Sun.COM
# ofroemel@prs-gmbh.de
# Uwe.Wiest@sun.com
# tunger@prs-gmbh.de
#
# Thanks to:
# stephan.figour@csfb.com, richard.turner@esa.int, marc_luescher@lotus.com,
```

```

# gerd.pruemm@alcatel.ch, meinrad.hodel@alcatel.ch,
#
#
# Description/Functionality:
# -----
#   Start/Stop of Lotus Partition Servers, all in one, or one by one.
#
#   Finds out how many Partition Servers are installed on the system,
#   by searching the system passwd table for users which look like:
#
#   "notes[0-9]".
#
#   Checks correct environment of all partitioned servers.
#
#
# Shortcuts:
# -----
#   LNSU == Lotus-Notes-Server-User
#
# Requirements/Assumptions:
# -----
#   1. needs Bourne Shell for the LNSU.
#   2. Partition Servers (LNSU's) must have the following form in the
#       system-passwd table: "notes" plus a one digit number.
#
#       e.g:"notes1"
#
#   3. if script is called with the server by server feature no
#       setup checking of the given domino-user will be done!
#   4. This script needs the environment variable: NOTESDATA_DIR to be
#       set in the .profile of the LNSU.
#   5. This script needs the environment variable: NSD_LOGDIR to be
#       set in the .profile of the LNSU.
#   6. The (empty) file ".hushlogin" in the Homedirectory of the LNSUS
#       must exist to prevent motd/mail-checking (look in /etc/profile).
#   7. The Variable "ServerKeyFileName" in the notes.ini must be set
#       to the correct ID-File of the Domino-Server (the same as KeyFileName)
#   8. The system should not be configured to run NIS. If so, you may have
#       to change
#       the lookups to /etc/passwd to the appropriate NIS-equivalents.
#       You may ask the authors for assistance.
#   9. LNSU's .profile should not be the default from /etc/skel
#       (strip off all unnecessary things if you have such). Only PATH,
#       NSD_LOGDIR, Notes_SHARED_DPOOLSIZE and other needed settings should
#       be included.
#   10. Put the script in /etc/init.d/lotus, make it executable and link it
#       like this:
#       cd /etc/rc3.d

```



```

# ln /etc/init.d/lotus S99lotus
# cd ../rc0.d
# ln /etc/init.d/lotus K00lotus
# cd ../rc1.d
# ln /etc/init.d/lotus K00lotus
#
# Tested Lotus releases:
# -----
# 4.6.3a, 4.6.6b, 4.6.6c, 4.6.7
# 5.0.1a, 5.0.3, 5.0.4, 5.0.4a, 5.0.5, 5.0.6, 5.0.6a
# 5.0.7, 5.0.7a, 5.0.8
#
# Versions:
# -----
# 1.0Klaus ZieglerFri May 7 09:15:28 MET DST 1999
# Basic start and stop procedure.
# 1.1Klaus ZieglerWed May 12 13:33:13 MET DST 1999
# Changed the way to shutdown a lotus server - using the lotus command
# "server -q" now, the kill -9 and PID procedure was to hard.
# 1.2Klaus ZieglerMon May 17 22:08:45 MET DST 1999
# Introduced variables: SERVER and LOTUSER.
# 1.3Klaus ZieglerWed May 26 23:30:55 MET DST 1999
# Made it possible to start one by one server, some echo cosmetics.
# 1.4Klaus ZieglerFri Jun 4 13:43:06 MET DST 1999
# Changed variable DIR into DATA and appended notes.ini to it, changed
# variable SERVER (using nawk and sed is a waste of cpu time).
# General cleanup and translation of comments from german to english.
# 1.5Klaus ZieglerWed Jun 23 16:32:49 MET DST 1999
# introduced functions: single_function, multiple_function.
# 1.6Klaus ZieglerWed Jun 23 16:51:20 MET DST 1999
# deleted function multiple_function and changed fuction single_function
# to function chkenv().
# 1.7Klaus ZieglerWed Jun 23 21:52:31 MET DST 1999
# Introduced function kill_server() to kill partition server,
# if Lotus shutdown has failed.
# 1.8Klaus ZieglerMon Jun 28 09:23:24 MET DST 1999
# more than one instance of the partition server didn't terminate
# correctly, fixed this bug.
# 1.9Oliver Froemel Tue Jun 29 15:54:52 MET DST 1999
# changed behavior of kill_server to use
# /opt/lotus/bin/tools/diag/nsd.sh -kill
# to have debugging-output to be used for analysis
# (needs environment-variable NSD_LOGDIR to be set in the users .profile)
# 2.0Klaus ZieglerThu Jul 15 13:54:54 MET DST 1999
# added feature to not stop the servers if they are already down.
# changed the echo output.
# added feature to use DOS-style notes.ini files.
# 2.1Klaus ZieglerThu Jul 15 14:50:29 MET DST 1999
# added wait-loop to wait until server comes down, if server does

```

```

# not come down it will be killed by function kill_server.
# 2.2Klaus ZieglerThu Jul 15 17:03:45 MET DST 1999
# modified script to make it possible to be called by cron, removed
# absolut pathnames from the programs, because variable PATH is set now,
# changed the way variable PROCS is build "ptree | wc -l | awk ..." zsss.
# 2.3Klaus ZieglerTue Jul 20 19:23:38 MET DST 1999
# deleted bug which shut down all notes-servers, if the given username
# was not correct. Added much setup checking.
# 2.4Klaus Ziegler Tue Jun 6 16:36:51 MET DST 2000
# introduced variable LOTUSSERVERS, to make script more flexible.
# Build loop to evaluate variable LOTUSSERVERS.
# 2.5Klaus Ziegler Tue Jun 6 19:26:30 MET DST 2000
# introduced Shortcuts colum at the top.
# modified function chkenv:
# checking for unique homedirectory of LNSU in system password database.
# checking for use of Bourne Shell for LNSU in system password database.
# checking for existing homedirectory for LNSU.
# checking for existing ".hushlogin" file in homedir of LNSU.
# checking for existing ".profile" file in homedir of LNSU.
# 2.6Klaus Ziegler Wed Jun 7 15:44:44 MET DST 2000
# Changed variable STR to be "ServerKeyFileName", comment changes,
# made the start/stop loops work with the variable LOTUSSERVERS.
# forced the variable SERVER to be lowercase letters, checking
# existing string ServerKeyFileName in notes.ini of the LNSU.
# Merged changes made by Oliver Froemel.
# 2.7Klaus Ziegler Thu Jun 8 18:08:15 MET DST 2000
# introduced function instcheck() and build into the start/stop switches,
# commented all functions.
# 2.8Klaus Ziegler Tue Jun 13 17:33:43 MET DST 2000
# Renamed function instcheck() back to chkenv.
# restructured function id_wrapper() and moved much parts into
# function chkenv(), modifyied status output, introduced function:
# users(), added variable CHECK. Corrected comments for functions.
# Added "for i loop" and "case switch" in function chkenv().
# Finished functionality to discover and start/stop all installed
# partition servers, so no more need for static variables !!!
# Much much testing.
# 2.9Oliver Froemel Wed Jun 28 12:51:18 MET DST 2000
# Modified function kill_server() to call
# /opt/lotus/bin/tools/diag/nsd.sh with the -batch-option
# to make kill-procedure work when called via cron
# 3.0Klaus Ziegler Thu Aug 17 15:49:14 MET DST 2000
# added checking of startup for the given partition server.
# if a partition-server is already running, it will not be re-started.
# created function chkrun() for this purpose, cause checking is
# peformed during general startup of all partitioned servers and for
# the one by one functionality of this script.
# 3.1Klaus Ziegler Wed Oct 18 19:14:20 MEST 2000
# Added the functionality to confirm the shutdown of a single

```

```

# partitioned server.
# 3.2Klaus Ziegler Thu Oct 19 10:31:32 MET DST 2000
# The functionality of fix 3.1 did not work if called via trusted
# remote shell for a single partition server. Also introduced switches
# -ps -memcheck -lsof to the call of the nsd.sh script from lotus.
# 3.3Klaus Ziegler Thu Oct 26 15:12:32 MET DST 2000
# removed the nsd.sh options: -ps -memcheck -lsof -- they caused the
# server not stop any more.
# 3.4Klaus Ziegler Thu Dec 14 15:52:34 MET 2000
# increase the timeout to shut the servers down to 5 minutes,
# cause the given 3 minutes were too short to shut down 1200 threads.
# 3.5Oliver Froemel Wed Feb 21 12:43:20 MET 2001
# Added functionality for a local Domino Admin Console
# Syntax to get it:
# cat >> cinput | tail -f coutput (put this in a script...)
# Attention: Check all the file-permissions carefully (and the umask).
# Otherwise, you may end up with a security breach from this.
# Removed confirmation of stopping the server (problems with using in
# cron). If you want this, ask the authors for a modified 3.5-version.
# Temporarily changed the DOSENFILE-Routine to look for \0x13
# We might have to change this again if we get hold of DOSENFILES from M$
# again. It is not wise to edit notes.ini from DOS and ftp it over at all.
# Admins should always do a "dos2unix" with the file afterwards.
# Next version may contain a forced "dos2unix" with notes.ini.....
# Corrected minor spelling mistakes.
# 3.6Oliver Froemel Tue Apr 3 19:11:01 MEST 2001
# Implemented function check_dosenfile because grep failed
# Now, this function can be called to check notes.ini and let it be
# converted automatically if any DOS-ish End-Of-Line-Combinations are
# found Replaced all other occurrences of DOSENFILE-checking by calls to
# this function Added some more comment-lines.
# 3.7Klaus Ziegler Fri Aug 10 10:52:03 MEST 2001
# If no LNSU are setup up the script prints out the error message:
# cat: cannot open /tmp/start_stop_server, this error is corrected by
# this version. If there are no LNSU users set up jet inform the
# admin/user to read the comments in this script.
# Modified function id_wrapper to be aware of capital ".ID" prefix
# of the ServerKeyFileName setting in notes.ini, merged both awk
# commands of funtion id_wrapper into one.
# 3.8Uwe Wiest Mon Sep 10 22:44:02 MEST 2001
# Changed the script to work with a seperate notesdata directory from the
# homedirectory of the unix user.
# This directory can be specified by adding the variable
# NOTESDATA_DIR to the .profile of the LNSU.
# Changes are recommended in the IBM Redbook "Domino on Solaris 8"

```

```
PATH=/usr/bin:/bin:/usr/proc/bin
```

```
#####
#
# check_dosenfile():
# This function is used to check if the notes.ini contains the DOS-ish
# line-feed+carriage-return-combination to signal end-of-line
# If any of these are found, the command "dos2unix" is called to automagically
# convert notes.ini to Unix-format.
#####
#

check_dosenfile()
{
    DOSENFIL=`nawk '/\r$/ { print NR }' $DATA | wc -m`
    if [ $DOSENFIL -gt 0 ]; then
        echo "\nConverting $DATA from DOS to Unix format...\c"
        dos2unix $DATA $DATA
        echo "- Done."
    fi
}

#####
#
# chkpid():
# Function chkpid() is used to check out if there are any remaining processes
# left for a given user.
# The function is always called after a "server -q" command, which is the usual
# way to shut down Lotus Domino server/servers. After the function is started
# it sets a timer (CNT=300 (5 min.)), a loop is started which waits for one
# second and increases variable NUM, if variable NUM reaches the value of
# variable CNT the function ends. Function chkpid() is also terminated if
# variable PROCS equals to zero, the "break" action of the "else" statement in
# the test argument is executed, which breaks the previous started
# "while do done" loop.
# The variable: $1 in this function always refers to the Lotus-Notes-User
# which runs this Domino-Server, whether it is partitioned or not.
#####
#

chkpid()
{
    echo
    CNT=300
    NUM=0
    while [ $CNT -ne $NUM ]; do
        PROCS=`ptree $1 | awk 'END{print NR}'`
        if [ $PROCS -gt 0 ]; then
            NUM=`expr $NUM + 1`
            sleep 1
        else

```

```

        break
    fi
done
}
#####
#
# chkenv():
# The function chkenv() is used to check installation environment and
# to set variables which are needed for status output.
# The following environment checks are done by chkenv() in the
# following order:
# 1. check for existing homedirectory for the LNSU.
# 2. check if homedirectory of LNSU is unique in system passwd.
# 3. check if the NOTESDATA_DIR is set correctly in the .profile
#    and if the NOTESDATA_DIR exists.
# 4. check if LNSU uses Bourne Shell.
# 5. check for existing .hushlogin file in homedirectory of LNSU.
# 6. check for existing .profile file in homedirectory of LNSU.
# 7. check for existing notes.ini file in homedirectory LNSU.
# 8. check if variable NSD_LOGDIR is set in the environment of the LNSU,
#    this is needed for the shutdown process.
# 9. check to see if the notes.ini file is in DOS/Windows format and set
#    variable DOSENFILE.
# 10. check to see if there is a ServerKeyFileName setting in notes.ini
#     of the LNSU.
# 11. create temporary file to start the servers.
#
# The variable: $1 in this function always refers to the Lotus-Notes-User
# which runs this Domino-Server, whether it is partitioned or not.
#####
#

chkenv()
{
    UNIXHOME=`awk -F: '/^'$1'/ {print $6}' /etc/passwd`

    NOTESDATA_TRUE=`su - $1 -c "env | grep -c NOTESDATA_DIR"`
    if [ $NOTESDATA_TRUE -eq 0 ]; then
        echo $ERSTR >>/tmp/run.$1.$$
        MESSAGE="No NOTESDATA_DIR variable in .profile of user: $1"
        echo "\n$MESSAGE\n" >>/tmp/run.$1.$$
        NOTESHOME="NOT SET FOR THIS USER!"
    else
        NOTESHOME=`su - $1 -c "env | grep NOTESDATA_DIR" | awk -F= '{print $2}'`
    fi

    ERSTR="===== ERROR ====="
    for i in home notesdata unique shell hush profile notes.ini
    do

```

```

case $i in
home)MESSAGE="no homedirectory at all for user: $1"
TEST="! -d $UNIXHOME" ;;
notesdata)MESSAGE="no notesdata directory at all for user: $1"
TEST="! -d $NOTESHOME";;
unique)HOMECHECK=`grep -c "/"$1" /etc/passwd`
MESSAGE="no unique homedirectory for user: $1"
TEST="$HOMECHECK -ne 1" ;;
shell)SHCHECK=`sed -n -e "s-^$1:.*:--p" /etc/passwd`
MESSAGE="user: $1 does not use a Bourne Shell"
TEST="$SHCHECK != /bin/sh" ;;
hush)MESSAGE="no .hushlogin file in homedirectory of user: $1"
TEST="! -f $UNIXHOME/.hushlogin" ;;
profile)MESSAGE="no .profile file in homedirectory of user: $1"
TEST="! -f $UNIXHOME/.profile" ;;
notes.ini)MESSAGE="no notes.ini file in notesdata directory of user: $1"
TEST="! -f $NOTESHOME/notes.ini" ;;
esac
if [ $TEST ]; then
echo $ERSTR >>/tmp/run.$1.$$
echo "\n$MESSAGE\n" >>/tmp/run.$1.$$
VALID=false
fi
done
if [ $VALID = true ]; then
NSD_TRUE=`su - $1 -c "env | grep -c NSD_LOGDIR"`
if [ $NSD_TRUE -eq 0 ]; then
echo $ERSTR >>/tmp/run.$1.$$
MESSAGE="No NSD_LOGDIR variable in .profile of user: $1"
echo "\n$MESSAGE\n" >>/tmp/run.$1.$$
fi
DATA="$NOTESHOME/notes.ini"
check_dosenfile $DATA
CHECKSTR=`grep -c ServerKeyFileName $NOTESHOME/notes.ini`
if [ $CHECKSTR -eq 0 ]; then
echo $ERSTR >>/tmp/run.$1.$$
MESSAGE="no string: ServerKeyFileName in notes.ini of user: $1"
echo "\n$MESSAGE\n" >>/tmp/run.$1.$$
fi
fi
if [ -f /tmp/run.$1.$$ ]; then
echo " FAILED see /tmp/run.$1.$$"
else
echo " -- Okay to run Domino."
echo "$1" >>/tmp/start_stop_server
fi
}

```

```
#####
#
# function chkrun() checks if a Domino Partition Server is already running,
# and sets a variable if it is not running.
# This function is needed to avoid starting a partitioned server twice.
# It is only called when using the script without any notesx-commandline-option
#####
#

chkrun()
{
    ISRUN=`ps -fU ${1} | awk '/server/ && $3==1 { print $2 }'`
    if [ $ISRUN ]; then
        echo "Domino Partition Server for user: ${1} already runs on PID: $ISRUN"
        START=NO
    else
        START=YES
    fi
}

#####
#
# function id_wrapper() sets the SERVER variable and forces it to lower case
# letters. The notes.ini could also be MS-DOS/Windows style file-format because
# function check_dosenfile() is called inside
#####
#

id_wrapper()
{
    NOTESHOME=`su - $1 -c "env | grep NOTESDATA_DIR" | awk -F= '{print $2}'`
    DATA="$NOTESHOME/notes.ini"

    check_dosenfile $DATA

    CHECKSTR=`grep -c ServerKeyFileName $NOTESHOME/notes.ini`

    NSD_DIR=`su - $1 -c "env | grep NSD_LOGDIR" | awk -F= '{print $2}'`
    cd $NOTESHOME
    STR=ServerKeyFileName
    SRV=`nawk -F= '/^'$STR'/{sub(".id","");sub(".ID","");print tolower($2)}' $DATA`
    SERVER=$SRV
}

#####
#
# loop():
# The function loop() is used to check the environment and to set
```

```

# variables. The variable LOTUSSERVERS is built at the startup of program.
# The for loop evaluates all the LNSU's which are found in the system password
# database and checks for their consistency via function: chkenv().
#####
#
loop()
{
    if [ $CHECK = true ]; then
        for i in $LOTUSSERVERS
        do
            echo "Checking environment for Domino-User: \c"
            echo "$i\c"
            chkenv $i
        done
    else
        echo "$LOTUSSERVERS" >>/tmp/start_stop_server
    fi
}

#####
#
# kill_server():
# The function kill_server() is used to check that a Domino server went
# down gracefully. If the server went down and did not leave any remaining
# processes nothing happens; if not the switch-user command is executed and all
# remaining processes are killed with the standard domino script:
# /opt/lotus/bin/tools/diag/nsd.sh as the given LNSU user.
#
# Note:
# -----
# The LNSU is variable $2 in this function.

kill_server()
{
    PROCS=`ptree $2 | awk 'END{print NR}'`
    if [ "$PROCS" -gt 0 ]; then
        echo "Server: $SERVER did not go down gracefully"
        echo "Please check the log-files in $NSD_DIR/"
        su - $2 -c "cd \"$NOTESHOME\"; /opt/lotus/bin/tools/diag/nsd.sh -user $2
        -batch -kill"
    fi
}

#####
#
# look how much Lotus-Notes-Partition-Servers are installed,
# they have to be in the systems passwd table to be useable.
# Remark: If you use NIS, you may have to change the appropriate files...

```



```

users()
{
    LOTUSSERVERS=`nawk '/^notes[0-9]:/ {sub(":.*", ""); print $0}' /etc/passwd`
}

#####
#

#####
#

# remove temporary old start/stop file could be left somehow.
[ -f /tmp/start_stop_server ] && rm -f /tmp/start_stop_server

#####
#
#
# Start/stop Lotus partition servers. All in one, or one by one.
#
#####
#
case "$1" in
start)# check if a single instance should be started
if [ $2 ]; then
    # get the Domino-Servername via function id_wrapper()
    id_wrapper ${2}
    chkrun ${2}
    if [ $START = YES ]; then
        echo "Starting Lotus Partition Server: $SERVER in background"
        # ensure that the input/output-files are cleared when restarting
        su - $2 -c "cd \"$NOTESHOME\"; cat /dev/null > cinput; cat /dev/null >
coutput"
        # now issue the start-command
        su - $2 -c "cd \"$NOTESHOME\"; /opt/lotus/bin/server <cinput>coutput"
>/dev/null 2>&1 &
    fi
else
    # get the Domino-server-names via functions:
    # users(), loop() and check their environment via chkenv().
    # Note: function loop() only calls function chkenv() if variable
    # CHECK has the value "true" in it.
    CHECK=true
    echo "General Domino Server/Servers Startup."
    echo "Looking for Domino-Server(s): \c"
    # if no LNSU user(s) are already setup, just give a message and exit.
    NOLNSU=`grep -c "^notes[0-9]" /etc/passwd`

```

```

if [ $NOLNSU = 0 ]; then
    echo "none found, please read comments in this script."
    exit 0
fi
# call function users to let /etc/passwd be examined for notes-users
users
echo $LOTUSSERVERS
VALID=true
loop
# start every domino-server which is set by function: chkenv().
if [ -f /tmp/start_stop_server ]; then
    for i in `cat /tmp/start_stop_server`
    do
        chkrun ${i}
        if [ $START = YES ]; then
            echo "Starting Lotus Partition Server: \c"
            id_wrapper ${i}
            echo "$SERVER in background\c"
            # ensure that the input/output-files are cleared prior to restarting
            su - ${i} -c "cd "$NOTESHOME"; cat /dev/null > cinput; cat /dev/null >
coutput"
            # now, do the actual starting of the server
            su - ${i} -c "cd "$NOTESHOME"; /opt/lotus/bin/server <cinput>coutput"
            >/dev/null 2>&1 &
            echo " - Done."
        fi
    done
    rm -f /tmp/start_stop_server
fi
fi ;;
stop)# get the Domino-Servername via function id_wrapper()
if [ $2 ]; then
    id_wrapper ${2}
    PROCS=`ptree $2 | awk 'END{print NR}'`
    if [ $PROCS -gt 0 ]; then
        echo "\nStopping Lotus Partition Server: $SERVER\c"
        su - $2 -c "cd "$NOTESHOME"; /opt/lotus/bin/server -q" >/dev/null &
        chkpid $2
        kill_server $SERVER ${2}
    else
        echo "Partition Server: $SERVER was not running !"
    fi
else
    # get the Domino-Servername names via function loop().
    CHECK=false
    users
    loop
    for i in `cat /tmp/start_stop_server`
    do

```

```

id_wrapper ${i}
PROCS=`ptree ${i} | awk 'END{print NR}'`
if [ $PROCS -gt 0 ]; then
    echo "Stopping Lotus Partition Server: $SERVER\c"
    su - $i -c "cd "$NOTESHOME"; /opt/lotus/bin/server -q" >/dev/null &
    echo " "
    chkpid $i
    kill_server $SERVER $i
else
    echo "Partition Server: $SERVER was not running !"
fi
done
[ -f /tmp/start_stop_server ] && rm -f /tmp/start_stop_server
echo "Lotus Partition Server shutdown complete !"
fi ;;
*)
echo "Usage: /etc/init.d/lotus { start | stop }\n"
echo "If you want to start/stop a single lotus instance:\n"
echo "Usage: /etc/init.d/lotus { start notes1 | stop notes1 }\n" ;;
esac

```

Domino IP resolve process

This appendix discusses the Domino server name to IP address resolve process and how the Notes name service can assist the process.

When a new workstation is set up for the first time with Notes, the protocol's Name to Address resolve process is used exclusively where the server's common name is expected to be resolvable. In most cases the client and the server are located within the same IP domain and IP domain level and can access a DNS server.

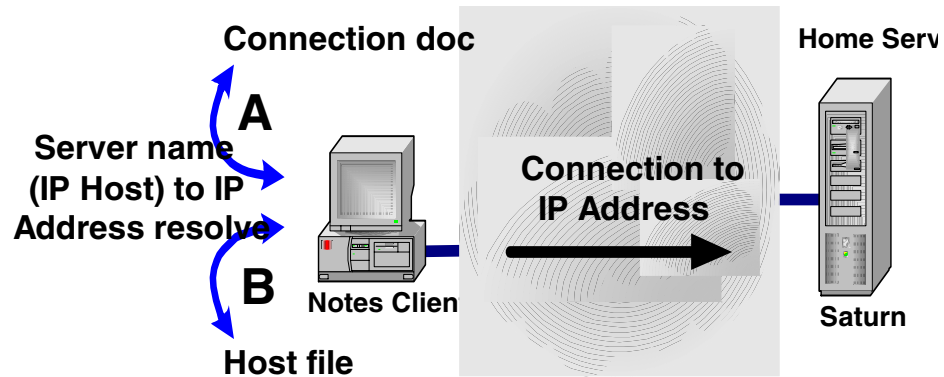


Figure G-1 Local protocol resolve

In some cases one does not have a DNS available or the Domino server is not listed in the DNS. In other cases the IP domain level where the Domino server is located is deeper than the client's IP domain level. In all of these cases, one can leverage a local host file or (if prepopulated in the user's personal address book via the personal NAB template) connection documents to guide the workstation to the Domino server. This approach must be used if there is no DNS server or the Domino server is not listed in the DNS (as one is likely to do with Internet-based resources). There are a few alternatives with hierarchical DNS resolve problems that require less effort, which we cover later on.

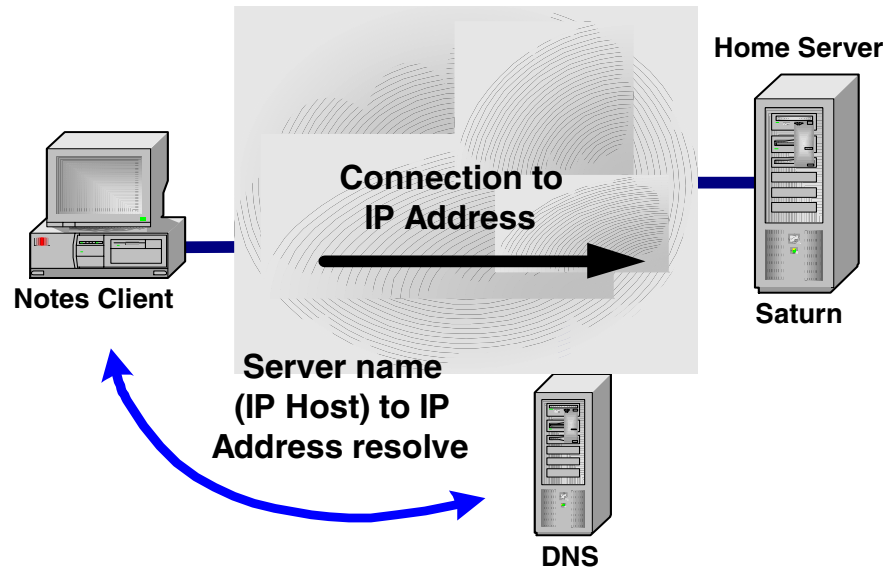


Figure G-2 Centralized protocol resolve

The order of resolve sometimes becomes important when you have multiple listings. A connection document is always used first, then the workstation's host file (default per the IETF RFCs). Because connection documents can be biased to a given location document, this allows one to have multiple listings for the same Domino server by different connection doc or host file entries. As an example, a user's laptop needs to access the same Domino server from the Internet as well as from the internal network, but the Domino server is known by different IP addresses either because there is a network address translation (NAT) Router in the pathway or the Domino server is Multi-homed with two interfaces.

Either directly placing the IP addresses within the connection documents, or placing two different host names which are only known within the connection documents to query either a Host file or DNS, allows the Admin to leverage either Notes setup profiles or user policies or replication of connection documents. Or by using OS replication services to maintain the host file entries directly.

So far our discussion has been how the native IP name resolve process works. If you have a UNIX, AS/400, or S/390 server this is what you expect. With Windows NT/2000/XP systems you may have a second or third name service present which could confuse things (i.e. NetBIOS). Ideally, the server's name should not be the same as the system's name, so name ghosting between the name spaces does not take place. Many connection failures or strange access reactions can be traced to NetBIOS name services getting in the way.

Leveraging the Notes name services

So far we have talked exclusively about how the Notes client uses the protocol's name services directly to gain access to the user's home server. Once there, the user's home server can offer assistance leveraging the Notes name service within the same Domino domain. For the Notes name service (NNS) to be effective, the given server's Server document must have a resolvable name within the protocol's name service. Depending on what your needs are you may require a *Hard* versus *Soft* resolve. A *Hard* resolve implies the full DNS hierarchy is listed out (i.e. red.boston.acme.com). A *Soft* resolve is when only the IP host name is listed (i.e. red). In most cases the Domino server's common name should be used as the simple host name in either case (i.e. 'red' and in the case of the Domino server called Red/Bos/Acme). One problem you can encounter is with Hard resolve when you have multiple IP addresses (Multi-homed system), unless you have a means to isolate the resolve from the different directions using different resolvers (host file or parallel DNS offer, as in the case of a shadow DNS).

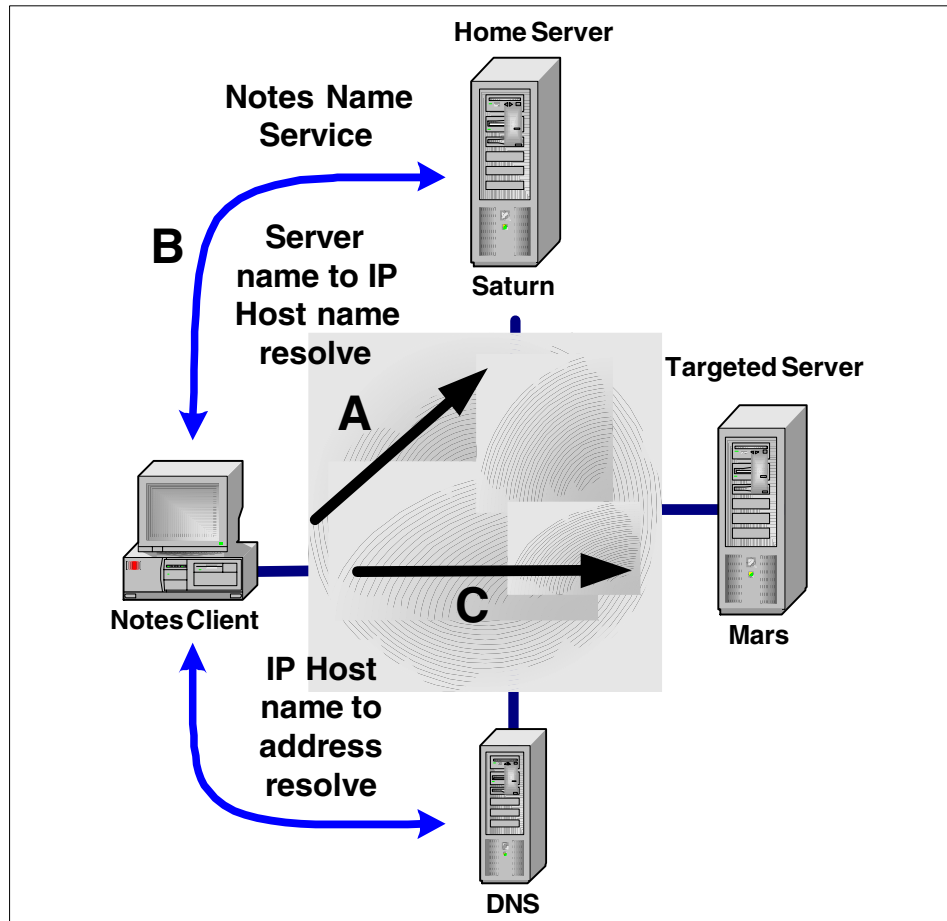


Figure 13-6 Notes name service to central resolve

Here we can see how the user's home server can offer an alternative host name or the fully qualified host name of the targeted server. So, looking at the example, we can see how the server Saturn is able to supply the Notes client the protocols name (i.e. mars.chicago.acme.com), and if the user's workstation's name lookup scope was based in 'boston.acme.com' the workstation is able to get to the correct subdomain level within the DNS offerings to locate the server in the 'chicago.acme.com' subdomain.

Leveraging a common secondary Notes name server

So far we have assumed the user's workstation was within the same DNS subdomain as the user's home server, so simple IP Host name resolve was possible for the server's common name (i.e. 'saturn'). In larger enterprise networks this is not possible, or the user's workstation may be accessing different DNS domains due to DHCP leases when they move about from site to site. While having a connection document for the user's home server can be very useful in making sure the Notes client can locate the home server, there are many good reasons not to do this with a large number of client installs. A better means is to leverage a single Domino server as the secondary name server. As long as the clients can locate this server by its simple IP host name (because it is located within the root level of the DNS domain), this allows a single management point to maintain the lookup offers by the Notes name service.

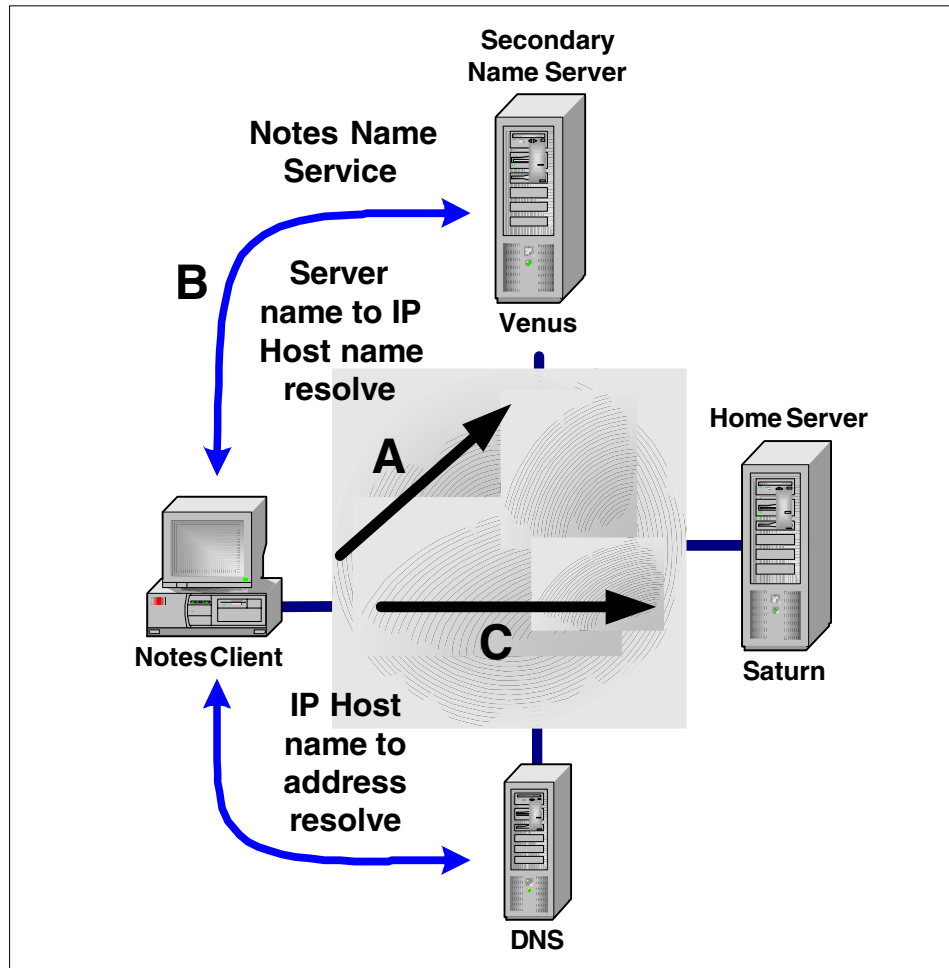


Figure G-3 Notes name service secondary server to central resolve

Here we can see how Venus, which is located within the root level of the acme IP domain (venus.acme.com), is accessible from any subdomain location within the acme domain, as long as each of the Server document's Net Address field offers the fully qualified host name of each Domino server Notes name. The name to address resolve is then a matter of the Domino Directory and DNS servers having accurate entries. Sometimes there is no network pathway to this secondary server from one location or another. This is where multiple listings of different servers by the same IP Host name can fill the gap. Additionally, having a single server can be thought to be risky, and this is where a Notes cluster can be useful. In the case of multilisting of the same IP Host name within the DNS, this requires a DNS based on BIND 9.xx to be functional. It has the advantage of *not*

requiring a sustained network connection between the Domino servers, but requiring only that the DNS tables are held in synch between the remote sites and the master DNS server system for the root and subdomains, and that the Domino directory is replicated as well.

Using Secondary name servers to back up the user's home server

One more use for secondary name servers is when the user's home server goes down or for some other reason is not accessible. This could be because the user is not at a location to gain access to his home server but can access other Domino servers within the Domino domain. As the following example shows, the user can still locate the targeted server 'Mars' even though his home server can't be reached for Notes name service resolve assistance.

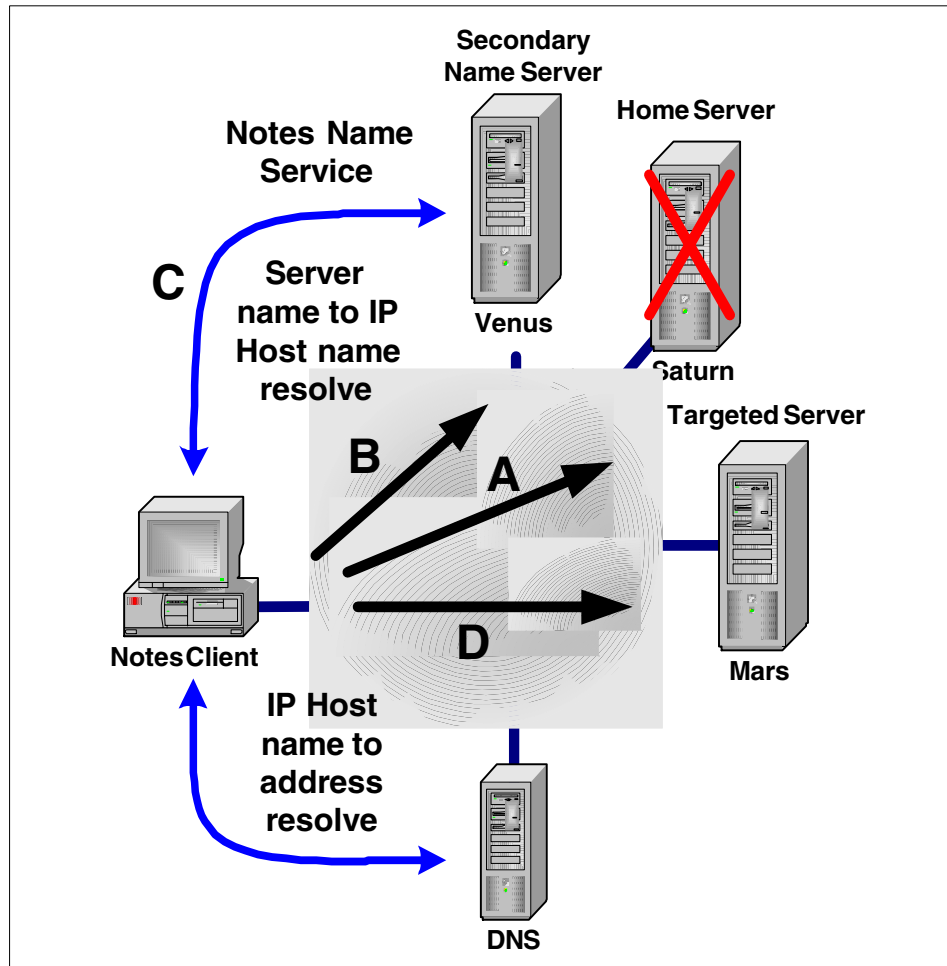


Figure 13-7 Secondary name service failover

Using a pass-through server

Unlike direct connections, one can encounter network requirements that can be confusing from a resolve process perspective when using a pass-through server since one quite often has to pass through to a different Domino domain. If the pass-through server has no means to query the inner domain's Domino directory or the protocol's name to address resolve is likewise restricted (both common practices within a DMZ designs, either the connecting Notes client has to tell the pass-through server the targeted system's IP address or the pass-through server needs to have connection documents to leverage on behalf of the connecting

Notes client as a the resolving proxy. While our focus is on Notes clients here, Domino servers from other Domino domains likewise can use the same processes. Of course authentication, certification, and ACL controls also play a part here, securing your Domino servers and their data.

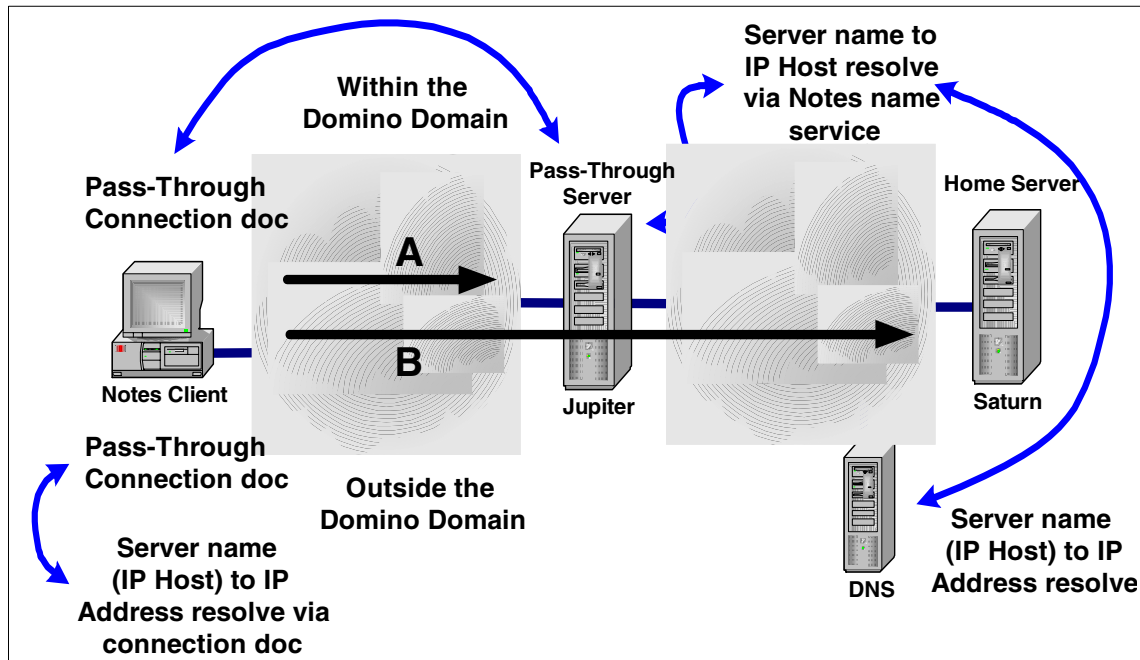


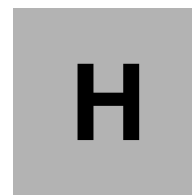
Figure 13-8 NRPC Pass-through

When using a pass-through server, the user's location document should point to the pass-through server as its secondary Notes name server. In our example we used a discrete server. Ideally, you should have a Notes cluster and point to the clusters name. Then either using a reverse proxy (Internet) or discrete connection documents for each of the clustered servers allows the remote system the ability to access any one of the servers within the cluster as its pass-through server.

Conclusions

In all of the configurations we have described here, the given server's Server document Net Address fields carry a fully qualified host name (FQHN), and if there is more than one NRPC TCP port, each offers the same name reference with the expectation that either the remote system's access by name resolve aims them to the correct interface's IP address, or, they have connection documents within Notes that bypass the protocol's name services. This is where

the overall design of your Domino and IP networks needs to be reviewed carefully to use the best methods given the requirements imposed. Your aim should be to minimize the creation of connection documents within the user's address book. At least limit the connection documents for the user's home server and, if required, the pass-through server, if you don't have a root-level secondary notes name server within your IP domain that all of your Notes clients can leverage if you have a hierarchical IP domain.



Additional material

This redbook refers to additional material that can be downloaded from the Internet as described below.

Locating the Web material

The Web material associated with this redbook is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

<ftp://www.redbooks.ibm.com/redbooks/SG245969>

Alternatively, you can go to the IBM Redbooks Web site at:

ibm.com/redbooks

Select the **Additional materials** and open the directory that corresponds with the redbook form number, SG245969.

Using the Web material

The additional Web material that accompanies this redbook includes the following files:

<i>File name</i>	<i>Description</i>
sg245969.tar	All available sample material from this book

How to use the Web material

Create a subdirectory (folder) on your workstation, and untar the contents of the Web material tar file into this folder.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these ITSO publications see, “How To Get IBM Redbooks.”

- ▶ *Customizing Quickplace*, SG24-6000
- ▶ *Lotus Sametime 2.0 Deployment Guide*, SG24-6206
- ▶ *Inside the Lotus Discovery Server*, SG24-6252
- ▶ *iNotes Web Access*, SG24-6518
- ▶ *Using Domino Workflow*, SG24-5963
- ▶ *XML Powered by Domino*, SG24-6207
- ▶ *Lotus Sametime Application Development Guide*, SG24-5651
- ▶ *COM Together — with Domino*, SG24-5670
- ▶ *Lotus Notes and Domino Take Center Stage*, SG24-5630
- ▶ *Performance Considerations for Domino Applications*, SG24-5602
- ▶ *Lotus Domino Release 5.0: A Developer's Handbook*, SG24-5331
- ▶ *Connecting Domino to the Enterprise Using Java*, SG24-5425
- ▶ *LotusScript for Visual Basic Programmers*, SG24-4856
- ▶ *Developing Web Applications Using Lotus Notes Designer for Domino 4.6*, SG24-2183
- ▶ *Lotus Notes 4.5: A Developers Handbook*, SG24-4876
- ▶ *Lotus Solutions for the Enterprise, Volume 1. Lotus Notes: An Enterprise Application Platform*, SG24-4837
- ▶ *Lotus Solutions for the Enterprise, Volume 2. Using DB2 in a Domino Environment*, SG24-4918
- ▶ *Lotus Solutions for the Enterprise, Volume 4. Lotus Notes and the MQSeries Enterprise Integrator*, SG24-2217

- ▶ *Lotus Solutions for the Enterprise, Volume 5. NotesPump, the Enterprise Data Mover*, SG24-5255
- ▶ *Enterprise Integration with Domino for S/390*, SG24-5150

Other Lotus-related ITSO publications

The publications listed in this section may also be of interest:

- ▶ *A Roadmap for Deploying Domino in the Organization*, SG24-5617
- ▶ *The Three Steps to Super.Human.Software: Compare, Coexist, Migrate; From Microsoft Exchange to Lotus Domino, Part One: Comparison*, SG24-5614
- ▶ *The Three Steps to Super.Human.Software: Compare, Coexist, Migrate; From Microsoft Exchange to Lotus Domino, Part Two: Coexistence and Migration*, SG24-5615
- ▶ *Lotus Notes and Domino R5.0 Security Infrastructure Revealed*, SG24-5341
- ▶ *Lotus Notes and Domino: The Next Generation in Messaging. Moving from Microsoft Mail to Lotus Notes and Domino*, SG24-5152
- ▶ *Eight Steps to a Successful Messaging Migration: A Planning Guide for Migrating to Lotus Notes and Domino*, SG24-5335
- ▶ *Deploying Domino in an S/390 Environment*, SG24-2182
- ▶ *The Next Step in Messaging: Upgrade Case Studies on Lotus cc:Mail to Lotus Domino and Lotus Notes*, SG24-5100
- ▶ *Lotus Notes and Domino: The Next Generation in Messaging. Moving from Novell GroupWise to Lotus Notes and Domino*, SG24-5321
- ▶ *High Availability and Scalability with Domino Clustering and Partitioning on Windows NT*, SG24-5141
- ▶ *From Client/Server to Network Computing, A Migration to Domino*, SG24-5087
- ▶ *Netfinity and Domino R5.0 Integration Guide*, SG24-5313
- ▶ *Lotus Domino R5 for IBM RS/6000*, SG24-5138
- ▶ *Lotus Domino Release 4.6 on IBM RS/6000: Installation, Customization and Administration*, SG24-4694
- ▶ *High Availability and Scalability with Domino Clustering and Partitioning on AIX*, SG24-5163
- ▶ *Lotus Domino for AS/400: Installation, Customization and Administration*, SG24-5181

- ▶ *Lotus Domino for S/390: Installation, Customization & Administration*, SG24-2083
- ▶ *Lotus Domino for S/390 Performance Tuning and Capacity Planning*, SG24-5149
- ▶ *Porting C Applications to Lotus Domino on S/390*, SG24-2092
- ▶ *Managing Domino/Notes with Tivoli Manager for Domino, Enterprise Edition, Version 1.5*, SG24-2104
- ▶ *Measuring Lotus Notes Response Times with Tivoli's ARM Agents*, SG24-4787
- ▶ *Using Tivoli Storage Manager to Back Up Lotus Notes*, SG24-4534

Redbooks on CD-ROMs

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at <http://www.redbooks.ibm.com/> for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

Also of interest

Sun Performance and Tuning, Java and the Internet, by Adrian Cockcroft and Richard Pettit ISBN 0-13-095249-4, available at:
<http://www.sun.com/books/catalog/Cockcroft/index.html>

Special notices

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.



Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

e (logo)® 	Redbooks
IBM ®	Redbooks Logo 
IBM @server	OS/390
APPN	OS/2
AS/400	MQSeries
AT	PowerPC
AIX	RS/6000
CT	Sametime
Current	SP
cc:mail	S/390
Domino	SP2
DB2	Sysplex Timer
Lotus	System/390
Lotus Notes	Wave
iNotes	Wizard
Lotus iNotes	400
Lotus Sametime	RS/6000
Mobile Notes	WebSphere
Notes	XT
Netfinity	

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere., The Power To Manage., Anything. Anywhere., TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others.

Index

Symbols

/tmp Configuring 20

A

Administration 137
 ACLs 145
 cluster replication 145
 compacting 145
 configuration 149, 162
 files 144
 full text indexes 145, 156
 mail 147, 160
 people and groups 142
 quotas 145, 156
 replication 148, 159
 server analysis 161
 server console 147
 server lists 139
 setting preferences 140
 troubleshooting 163
 Web administrator 150
Admintool 25
Analysis tools 266
Application migration 37
 case sensitivity 40
 file permissions 39

B

Backup 281, 363
 backup and replication 285
 backup cycle 285
 backup features 289
 backup open files 282
 backup planning 302
 DBIID 287
 library unit 282
 Lotus C API 288
 transaction logging 286
 utilities 283
Billing 134
 agent class 134
 billing classes 134

BillingAddinRuntime 135
BillingAddinWakeup 135
BillingSuppressTime 135
BILLSES program 136
 configuration 134
 customizing 136
 database class 135
 document class 135
 how it works 134
 HTTPRequest class 135
 mail class 135
 replication class 135
 session class 135
 storing the information 135
 troubleshooting 136
 tuning 135
Bindsock 247
 suid 247

C

CASE tools 38
cconsole 165
CGI 243
 CGI Timeout 243
Chmod command 247
Cluster tuning 109
Clustered Servers
 database distribution 121
 server availability 122
Clustered servers 120
 cluster database cache 133
 cluster directory database 127
 creating a cluster 123
 debug 133
 enabling multiple cluster replicators 131
 etting up your cluster 130
 failover 123
 how many CLREPL tasks 130
 how many servers in a cluster 130
 logging 133
 removing a server from a cluster 127
 statistics 131
 troubleshooting 132

- workload balancing 121
- Clustering
 - maximum number of users 123
- Clusters 23
 - cluster administration 23
 - cluster manager 23
 - cluster replicator 23
 - hardware 24
 - tuning 109
- Compact
 - transaction logging 287
- Configuration checklist 24
- Cookie 235
 - HTTP_COOKIE 236
- Coreadm 10
- CPU
 - performance 93
- CSH shell 27

D

- DECS 3
- Devfsadm 11
- Df command 367
- Directory catalog 3
- Disks
 - configuring 19
 - file system layout 36
 - performance 94
- Domino advanced services 111, 137
- Domino assistance 3
- Domino character console 165
 - commands 168
 - starting 166
 - stopping 167
- Domino directory 3
- Domino log 266
 - Domino web log 266
- Domino server
 - shutting down 85
 - starting from command line 81
 - starting in the background 82
 - starting using a script 83
- Domsetup 377
- Dos2unix command 244
- DSAPI 232

E

- Enterprise Integration

- connectors 272
- contest tool 279
- lctest tool 278
- leiclean tool 280
- Lotus Enterprise Integrator (LEI) 273
- multi-value data 272
- ODBC 276
- Enterprise integration 272

F

- File system layout 36

G

- Gcc compiler 364
- Groups
 - creating 25

H

- HTTP 230

I

- Ifconfig command 115, 251
- Incremental backup 286
- Installing Domino
 - additional Domino servers 64
 - checklist 44
 - directory 48
 - first Domino server 57
 - installing server code 46
 - installing Solaris 44
 - partitioned servers 50
 - re-running setup 73
 - root not supported 57
 - server selection 48
 - starting the server 81
 - using domsetup 377
- Internet Cluster Manager 227
- Internet Cluster Manager (ICM) 258
 - ICM statistics 262
- lstat command 92

J

- Java 263
 - notesjre 263
 - servlets 265

L

Ldd command 280
Light Weight Processes 241

M

Memory
 performance 93
Memory dump 249
MinNewMailPoll 108, 109
Mmap() 20
Mpstat command 92

N

Ndd command 97
Netstat command 92, 96
Network configuration 30, 95
 network cards 32
Network tuning 228, 229
 CLOSE WAIT state 229
 ndd 229
 sq_max_size 229
Notes C API 363
Notes.ini 103
 BillingClass 134
notes.ini 230
 BillingAddinOutput 135
 DominoAsynchronizeAgents 244
 DominoNoBanner 234
 ICMDebugLevel 263
 RTR_Cached_Handle_Disable 133
 RTR_logging 133
 SERVER_AVAILABILITY_THRESHOLD 122
 SERVER_DEBUG_CLUSTERS 133
 SERVER_TRANSINFO_NORMALIZE 122
 ServerTasks 230

P

Partitioned Servers 112
 configuration 113
 installation 112
 multiple NICs 117
 Notes shared memory 119
 Notes_SHARED_DPOOLSIZE 119
 TCP/IP configuration 113
 troubleshooting 120
Partitions 22
 account IDs 29

 configuring servers 73, 74
 data directory 29
 home directory 29

Patches 44

Performance 241

 cluster tuning 109
 CPU 93
 disks 94
 file system tuning 95
 HTTP threads 241
 kernel tuning 95
 memory 93
 network configuration 95
 RAID 102
 semaphore timeouts 352
 Web server 109, 241

Pgrep command 11

Pkill command 11

Proc 10

Prtvtoc command 368

Ps command 92

R

RAID 102
Restore and backup 303
root 57
RSS (Resident Set Size) 249

S

Security 79, 169, 233
 Access Control List (ACL) 195
 access security 189
 anti-virus products for Domino 197
 at.deny and at.allow files 175
 automount service 181
 cron command 175
 crontab command 175
 Domino flat certificates 187
 Domino security 186
 file permissions 171
 file system 171
 file system partitions 172
 file systems to mount 172
 File Transfer Protocol (FTP) 180
 fix-modes script 171
 heirarchical certificates 187
 inetd.conf file 180
 inetsvc file 182

- init system 176
- logcheck perl script 186
- login file 178
- mount command 174
- ndd command 183
- Network File System (NFS) 181
- network service security 178
- Notes certificate 186
- Notes ID 187
- passmgmt command 174
- passwd command 175
- protecting a Domino server 188
- protecting access during setup 188
- remote access services 179
- remote execution service (rexec) 180
- restrict access 190
- restrict java 191
- restrict passthru access 191
- restrict server agents 190
- rmmount.conf file 174
- S/MIME 233
- SEAM 179
- secure shell (ssh) 185
- set-group-id 172
- set-user-id 172
- showrev command 171
- Solaris patches 170
- Solaris SANS security script 185
- sudo tool 184
- TCP wrapper package 184
- telnet 179
- titan 185
- user accounts 174
- vold daemon 173
- Web server security 192
- X.509 certificate 233
- Semaphore debug 353
- Semaphore timeouts 352
- Sendmail 182
- Server 233
- Services
 - disabling default 20
- Session authentication 234
- Sizing
 - CPU 13
- Statistics 244
 - Web statistics 244
- Swap
 - configuring 20
- Syslog 177, 379, 385
- SZ (Set Size) 249

T

- Telnet command 245
- Transaction logging 286
 - archive logging 287
 - circular logging 286
 - compact 287
 - DBIID 287
- Troubleshooting 163, 245, 305
 - .NOTESMEM_please_do_not_remove files 317
 - adb tool 329
 - ANSI tool 321
 - core dump 310
 - core file 326
 - corrupted databases 351
 - crash 306, 310, 332
 - database maintenance policies 341
 - database overcrowding rejections 344
 - Debug notes.ini variables 338
 - DEBUG_CAPTURE_TIMEOUT 338
 - DEBUG_ENABLE_CORE 327
 - DEBUG_OUTFILE 338
 - DEBUG_SHOW_TIMEOUT 338
 - DEBUG_THREADID 341
 - debugthreadlogging 247
 - disk space 319
 - Domino peak users 345
 - fatal_error() call 316
 - fault recovery 334
 - fault recovery script 335
 - faulting stack thread 333
 - files for support 309
 - freezing all server threads 306, 332
 - FTP support site 309
 - full text index problems 343
 - gcore command 328
 - hang 307, 311
 - how to prevent a crash 341
 - how to read a core file 329
 - http memory leaks 249
 - HTTP thread debugging 247
 - iostat command 308, 312
 - ipcrm command 350
 - IPCS 317
 - ipcs command 350

- Killprocess notes.ini variable 331
- LookupHandle
 - handle not allocated 340
- memcheck tool 323, 331
- memcheck.dump file 325
- memory dump 326
- memory mapped files, mmap 317
- mpstat command 308
- NSD 306, 312, 314
- NSD options 312
- NSD warnings/errors 320
- NSD_LOGDIR env variable 312
- NSF_DbCacheMaxentries 344
- open databases 325, 345
- performance problems 308, 312
- proctool 312
- pstack command 333
- req###.log 247
- semaphore timeouts 339, 344
- SEMDEBUG.TXT file 338
- server does not start 349, 351
- signal.h file 333
- snoop tool 348
- Solaris patches 320
- sotrust command 347
- startup and shutdown Domino script 335
- strings command 330
- swap info 319
- sysdef command 327
- threads spin 310
- truss command 346
- ulimit command 327
- vmstat command 308, 312, 320
- Web server 245
- Truss command 246
- Web server 227, 231
 - asynchronous Web agents 243
 - domlog.nsf 232
 - file permissions 240
 - file protection 238
 - hanging 246
 - httpd.cnf 247
 - input timeout 242
 - Maximum request over a single connection 231
 - maximum requests over a single connection setting 242
 - number of active threads 231
 - output timeout 243
 - protect directives 238
 - setting HTTP timeouts 242
 - tell commands 246
 - tell HTTP commands 249
 - URL mappings 256
 - URL redirection 257
 - virtual host 116, 252
 - virtual server 251
 - Web realms 237
 - Web stress tools 244
- What's new in Solaris 8 10
- Workload balancing 121

U

- Umask 177
- Users
 - admintool 27
 - creating 25

V

- Vmstat command 92

W

- Web administrator 154



Lotus Domino R5 for Sun Solaris 8

(0.5" spine)
0.475" <-> 0.875"
250 <-> 459 pages



Lotus Domino R5 for Sun Solaris 8

Updated for Domino R5.0.8 and Sun Solaris 8

Tuning Domino and Solaris

Sizing Solaris for Domino

This IBM Redbook tells you how to run Lotus Domino R5.0.8 on the Sun Solaris 8 Operating Environment. (It contains information that has been revised and updated from the previous edition, which addresses Domino R5.0.2a and Solaris 7.) While the Lotus Domino server is platform-independent, each platform it runs on requires some additional platform-specific knowledge and configuration in order to ensure it operates efficiently and at maximum capability.

The primary focus is to explain the installation, configuration, and performance tuning of Domino R5 in this environment. We take you through all the steps required to run a Domino R5 server on Solaris 8, from choosing the right hardware, installing Solaris and Domino, tuning the OS and the Domino server and performing administrative tasks, through to problem determination and troubleshooting.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks